

Сервисные маршрутизаторы серии ESR  
**ESR-15, ESR-15R, ESR-30, ESR-3200**  
Контроллеры беспроводного доступа  
**WLC-15, WLC-30, WLC-3200**

Справочник команд CLI  
Версия ПО 1.30

## Содержание

1 Введение .....	3
2 История изменений.....	5
3 Правила пользования командной строкой .....	23
4 Структура системы команд .....	28
5 Команды пользовательского интерфейса.....	55
6 Управление программным обеспечением и конфигурацией .....	79
7 Настройка TFTP-сервера .....	114
8 Настройка DNS .....	117
9 Настройка WLC.....	120
10 Настройка mDNS-reflector.....	379
11 Настройка общесистемных параметров .....	383
12 Управление системными часами.....	404
13 Настройка AAA .....	423
14 Конфигурирование и мониторинг интерфейсов .....	482
15 Управление L2-функциями .....	594
16 Работа с адресными таблицами.....	641
17 Настройка VRF .....	656
18 Настройка IP-адресации .....	660
19 Настройка IPv6-адресации .....	667
20 Управление профилями.....	681
21 Управление NAT .....	702
22 Управление тунелированием, VPN и удалённым доступом.....	728
23 Маршрутизация .....	897
24 Настройки MPLS.....	1121
25 Резервирование .....	1156
26 Безопасность .....	1254
27 Управление QoS .....	1394
28 Мониторинг и управление .....	1433
29 Настройка DHCP.....	1542
30 Настройка SLA .....	1596
31 Настройка контроля абонентов (BRAS).....	1628
32 Настройка SoftGRE контроллера туннелей.....	1662

# 1 Введение

- [Аннотация](#)
- [Целевая аудитория](#)
- [Примечания и предупреждения](#)
- [Используемые сокращения](#)

## Аннотация


В настоящем руководстве приведено описание команд CLI для администратора сервисного маршрутизатора серии ESR и контроллера беспроводного доступа WLC (в дальнейшем именуемого маршрутизатором/контроллером или устройством).


Интерфейс командной строки (Command Line Interface, CLI) – интерфейс, предназначенный для управления, просмотра состояния и мониторинга устройства. Для работы потребуется любая установленная на ПК программа, поддерживающая работу по протоколам Telnet, SSH или прямое подключение через консольный порт (например, HyperTerminal).

## Целевая аудитория

Справочник команд CLI предназначен для технического персонала, выполняющего настройку и мониторинг маршрутизатора серии ESR или контроллера беспроводного доступа WLC посредством интерфейса командной строки (CLI). Квалификация технического персонала предполагает знание основ работы стека протоколов TCP/IP, принципов построения Ethernet-сетей.

## Примечания и предупреждения

 Примечания содержат важную информацию, советы или рекомендации по использованию и настройке устройства.

 Предупреждения информируют пользователя о ситуациях, которые могут нанести вред программно-аппаратному комплексу, привести к некорректной работе системы или потере данных.

## Используемые сокращения

- AAA – Authentication, Authorization, Accounting
- ARP – Address Resolution Protocol
- BGP – Border Gateway Protocol
- BRAS – Broadband Remote Access Server
- DHCP – Dynamic Host Configuration Protocol
- DNS – Domain Name System
- DDoS – Distributed Denial of Service
- DoS – Denial of Service
- FIFO – First In, First Out
- FTP – File Transfer Protocol
- FXO – Foreign eXchange Office
- FXS – Foreign eXchange Subscriber
- GRE – Generic Routing Encapsulation
- GRED – Gentle Random Early Detaction
- HTTP – HyperText Transfer Protocol
- HTTPS – HyperText Transfer Protocol Secure
- ICMP – Internet Control Message Protocol
- IKE – Internet Key Exchange

- IDS – Intrusion Detection System
- IP – Internet Protocol
- IP4IP4 – IP in IP
- IPS – Intrusion Prevention System
- IPsec – IP Security
- L2TP – Layer 2 Tunneling Protocol
- L2TPv3 – Layer 2 Tunneling Protocol version 3
- LACP – Link Aggregation Control Protocol
- LAG – Link Aggregation Group
- LDAP – Lightweight Directory Access Protocol
- LLDP – Link Layer Discovery Protocol
- MAC – Media Access Control
- MTU – Maximum Transmission Unit
- mDNS – Multicast Domain Name System
- NAT – Network Address Translation
- NAS – Network Access Server
- NTP – Network Time Protocol
- OSPF – Open Shortest Path First
- PPP – Point-to-Point Protocol
- PPTP – Point-to-Point Tunneling Protocol
- PPPoE – Point-to-point protocol over Ethernet
- PSK – Pre-Shared Key
- QoS – Quality of Service
- RADIUS – Remote Authentication Dial In User Service
- RED – Random early detection
- RIP – Routing Informational Protocol
- SFTP – Secure Shell File Transfer Protocol
- SIP – Session Initiation Protocol
- SLA – Service Level Agreement
- SNMP – Simple Network Management Protocol
- SCP – Secure Copy Protocol
- SP – Strict Priority
- SSH – Secure Shell
- STP – Spanning Tree Protocol
- TACACS – Terminal Access Controller Access Control System
- TFTP – Trivial File Transfer Protocol
- URL – Uniform Resource Locator
- VLAN – Virtual Local Area Network
- VPN – Virtual Private Network
- VRF – Virtual Routing/Forwarding
- VRRP – Virtual Router Redundancy Protocol
- VTI – Virtual Tunnel Interface
- WAN – Wide Area Network
- WINS – Windows Internet Name Service
- WRR – Weighted Round Robin
- XAUTH – eXtended Authentication

## 2 История изменений

Версия документа	Дата выпуска	Содержание изменений
Версия 1.34	07.02.2025	Изменения в разделах: <a href="#">Правила пользования командной строкой</a> <a href="#">Управление программным обеспечением и конфигурацией</a> <a href="#">Настройка WLC</a> <a href="#">Настройка WEB-сервера</a> <a href="#">Управление профилями ap-models</a> <a href="#">Управление NAT</a> <a href="#">Настройка общих параметров для failover-сервисов</a> <a href="#">Настройка SoftGRE контроллера туннелей</a>
Версия 1.33	20.01.2025	Изменения в разделах: <a href="#">Правила пользования командной строкой</a>
Версия 1.32	19.08.2024	Изменения в разделах: <a href="#">Настройка WLC</a> <a href="#">Настройка RADIUS-сервера</a> <a href="#">Настройка WEB-сервера</a> <a href="#">Работа с сертификатами</a> <a href="#">Управление профилями ap-models</a>

Версия документа	Дата выпуска	Содержание изменений
Версия 1.31	15.05.2024	<p>Добавлен раздел:</p> <p><a href="#">Настройка Crypto-Sync</a></p> <p>Изменения в разделах:</p> <p><a href="#">Правила пользования командной строкой</a></p> <p><a href="#">Структура системы команд</a></p> <p><a href="#">Команды пользовательского интерфейса</a></p> <p><a href="#">Управление программным обеспечением и конфигурацией</a></p> <p><a href="#">Настройка DNS</a></p> <p><a href="#">Настройка общесистемных параметров</a></p> <p><a href="#">Управление системными часами</a></p> <p><a href="#">Настройка AAA</a></p> <p><a href="#">Конфигурирование и мониторинг интерфейсов</a></p> <p><a href="#">Конфигурирование и мониторинг туннелей</a></p> <p><a href="#">Управление L2-функциями</a></p> <p><a href="#">Работа с адресными таблицами</a></p> <p><a href="#">Настройка VRF</a></p> <p><a href="#">Настройка IP-адресации</a></p> <p><a href="#">Настройка IPv6-адресации</a></p> <p><a href="#">Управление профилями</a></p> <p><a href="#">Управление NAT</a></p> <p><a href="#">Настройки IPsec VPN</a></p> <p><a href="#">Управление VPN. Настройки удаленного доступа</a></p> <p><a href="#">Маршрутизация</a></p> <p><a href="#">Настройки MPLS</a></p> <p><a href="#">Резервирование</a></p> <p><a href="#">Безопасность</a></p> <p><a href="#">Управление QoS</a></p> <p><a href="#">Мониторинг и управление</a></p> <p><a href="#">Настройка DHCP</a></p> <p><a href="#">Настройка SLA</a></p>

Версия документа	Дата выпуска	Содержание изменений
Версия 1.30	13.10.2023	<p>Изменения в разделах:</p> <ul style="list-style-type: none"> <li><a href="#">Настройка AAA</a></li> <li><a href="#">Конфигурирование и мониторинг туннелей</a></li> <li><a href="#">Настройка IP-адресации</a></li> <li><a href="#">Настройки IPsec VPN</a></li> <li><a href="#">Общие команды анонсирования и приема маршрутов</a></li> <li><a href="#">Маршрутизация на основе политик (PBR)</a></li> <li><a href="#">Управление VRRP</a></li> <li><a href="#">Управление Firewall</a></li> <li><a href="#">Управление списками контроля доступа (ACL)</a></li> <li><a href="#">Управление системой предотвращения вторжений (IPS/IDS)</a></li> <li><a href="#">Управление SYSLOG</a></li> <li><a href="#">Настройка доступа SSH, Telnet</a></li> <li><a href="#">Настройка DHCP</a></li> <li><a href="#">Настройка Tracking</a></li> </ul>
Версия 1.29	09.10.2023	<p>Добавлены разделы:</p> <ul style="list-style-type: none"> <li><a href="#">Настройка Crypto-Sync</a></li> <li><a href="#">Настройка WEB-сервера</a></li> </ul> <p>Изменения в разделах:</p> <ul style="list-style-type: none"> <li><a href="#">Настройка WLC</a></li> <li><a href="#">Настройка AAA</a></li> <li><a href="#">Управление L2-функциями</a></li> <li><a href="#">Настройка резервирования Firewall/NAT</a></li> <li><a href="#">Настройка доступа SSH, Telnet, Web</a></li> <li><a href="#">Настройка SoftGRE контроллера туннелей</a></li> </ul>

Версия документа	Дата выпуска	Содержание изменений
Версия 1.28	31.03.2023	<p>Добавлены разделы:</p> <ul style="list-style-type: none"> <li><a href="#">Управление Dual-Homing</a></li> </ul> <p>Изменения в разделах:</p> <ul style="list-style-type: none"> <li><a href="#">Команды пользовательского интерфейса</a></li> <li><a href="#">Настройка TFTP-сервера</a></li> <li><a href="#">Настройка общесистемных параметров</a></li> <li><a href="#">Управление системными часами</a></li> <li><a href="#">Настройка AAA</a></li> <li><a href="#">Конфигурирование и мониторинг туннелей</a></li> <li><a href="#">Настройка VRF</a></li> <li><a href="#">Настройка IP-адресации</a></li> <li><a href="#">Управление профилями</a></li> <li><a href="#">Управление NAT</a></li> <li><a href="#">Настройки IPsec VPN</a></li> <li><a href="#">Общие настройки маршрутизации</a></li> <li><a href="#">Общие команды анонсирования и приема маршрутов</a></li> <li><a href="#">Маршрутизация на основе политик (PBR)</a></li> <li><a href="#">Настройка параметров протокола BFD</a></li> <li><a href="#">Настройка протоколов OSPF и OSPFv3</a></li> <li><a href="#">Настройки MPLS</a></li> <li><a href="#">Управление VRRP</a></li> <li><a href="#">Настройка резервирования DHCP</a></li> <li><a href="#">Настройка резервирования Firewall/NAT</a></li> <li><a href="#">Управление Firewall</a></li> <li><a href="#">Управление QoS</a></li> <li><a href="#">Управление SYSLOG</a></li> <li><a href="#">Настройка DHCP</a></li> </ul>
Версия 1.27	12.09.2022	<p>Изменения в разделах:</p> <ul style="list-style-type: none"> <li><a href="#">Конфигурирование и мониторинг туннелей</a></li> <li><a href="#">Мониторинг и управление -&gt; Настройка доступа SSH, Telnet</a></li> </ul>



Версия документа	Дата выпуска	Содержание изменений
Версия 1.26	07.06.2022	<p>Изменения в разделах:</p> <ul style="list-style-type: none"> <li>Команды пользовательского интерфейса</li> <li>Управление программным обеспечением и конфигурацией</li> <li>Настройка DNS</li> <li>Управление VPN. Настройки удаленного доступа</li> <li>Маршрутизация -&gt; Маршрутизация на основе политик (PBR)</li> <li>Маршрутизация -&gt; Настройка протокола BFD</li> <li>Маршрутизация -&gt; Настройка протоколов OSPF и OSPFv3</li> <li>Настройки MPLS</li> <li>Безопасность -&gt; Управление системой предотвращения вторжений (IPS/IDS)</li> <li>Мониторинг и управление -&gt; Управление Netflow</li> <li>Мониторинг и управление -&gt; Настройка доступа SSH, Telnet</li> <li>Настройка контроля абонентов (BRAS)</li> </ul> <p><b>Добавлены разделы:</b></p> <ul style="list-style-type: none"> <li>Настройка TFTP-сервера</li> </ul>
Версия 1.25	03.02.2022	<p>Изменения в разделах:</p> <ul style="list-style-type: none"> <li>Структура системы команд</li> <li>Управление программным обеспечением и конфигурацией</li> <li>Настройка общесистемных параметров</li> <li>Конфигурирование и мониторинг туннелей</li> <li>Управление L2-функциями</li> <li>Управление VPN. Настройки удаленного доступа</li> <li>Управление QoS</li> <li>Маршрутизация -&gt; Маршрутизация на основе политик (PBR)</li> <li>Мониторинг и управление -&gt; Управление SYSLOG</li> </ul>

Версия документа	Дата выпуска	Содержание изменений
Версия 1.24	15.10.2021	<p>Изменения в разделах:</p> <ul style="list-style-type: none"> <li>Правила пользования командной строкой</li> <li>Структура системы команд</li> <li>Команды пользовательского интерфейса</li> <li>Управление программным обеспечением и конфигурацией</li> <li>Настройка общесистемных параметров</li> <li>Настройка AAA</li> <li>Конфигурирование и мониторинг интерфейсов</li> <li>Конфигурирование и мониторинг туннелей</li> <li>Настройки IPsec VPN</li> <li>Настройка протокола BGP</li> <li>Настройка протоколов OSPF и OSPFv3</li> <li>Настройки MPLS</li> <li>Управление VRRP</li> <li>Настройка объектов отслеживания событий</li> <li>Управление системой предотвращения вторжений (IPS/IDS)</li> <li>Управление QoS</li> <li>Управление Netflow</li> <li>Настройка доступа SSH, Telnet</li> <li>Настройка SoftGRE контроллера туннелей</li> </ul>

Версия документа	Дата выпуска	Содержание изменений
Версия 1.23	29.06.2021	<p>Изменения в разделах:</p> <ul style="list-style-type: none"> <li>Структура системы команд</li> <li>Команды пользовательского интерфейса</li> <li>Управление программным обеспечением и конфигурацией</li> <li>Настройка общесистемных параметров</li> <li>Конфигурирование и мониторинг интерфейсов</li> <li>Управление L2-функциями</li> <li>Работа с адресными таблицами</li> <li>Управление профилями</li> <li>Настройки IPsec VPN</li> <li>Маршрутизация -&gt; Общие настройки маршрутизации</li> <li>Маршрутизация -&gt; Настройка протокола BFD</li> <li>Безопасность -&gt; Управление Firewall</li> <li>Безопасность -&gt; Управление логированием и защитой от сетевых атак</li> <li>Безопасность -&gt; Настройка системы предотвращения вторжений (IPS/IDS)</li> <li>Управление QoS</li> <li>Мониторинг и управление -&gt; Управление SYSLOG</li> <li>Мониторинг и управление -&gt; Настройка зеркалирования</li> <li>Настройка контроля абонентов (BRAS)</li> <li>Настройка SoftGRE контроллера туннелей</li> </ul>

Версия документа	Дата выпуска	Содержание изменений
Версия 1.22	29.10.2020	<p>Изменения в разделах:</p> <ul style="list-style-type: none"> <li>Структура системы команд</li> <li>Управление программным обеспечением и конфигурацией</li> <li>Настройка общесистемных параметров</li> <li>Управление системными часами</li> <li>Настройка AAA</li> <li>Конфигурирование и мониторинг туннелей</li> <li>Управление L2-функциями</li> <li>Работа с адресными таблицами</li> <li>Управление фильтрацией</li> <li>Управление логированием и защитой от сетевых атак</li> <li>Настройки IPsec VPN</li> <li>Управление VPN. Настройки удаленного доступа</li> <li>Маршрутизация -&gt; Настройка протокола BGP</li> <li>Маршрутизация -&gt; Настройка протокола IS-IS</li> <li>Настройки MPLS</li> <li>Резервирование</li> <li>Управление QoS</li> <li>Мониторинг и управление</li> <li>Настройка DHCP</li> <li>Настройка SLA</li> <li>Настройка контроля абонентов (BRAS)</li> <li>Управление системой предотвращения вторжений (IPS/IDS)</li> </ul>
Версия 1.21	05.06.2020	<p>Изменения в разделах:</p> <ul style="list-style-type: none"> <li>Структура системы команд</li> <li>Управление программным обеспечением и конфигурацией</li> <li>Конфигурирование и мониторинг туннелей</li> <li>Управление L2-функциями</li> <li>Управление Firewall</li> <li>Управление логированием и защитой от сетевых атак</li> <li>Управление NAT</li> <li>Настройки IPsec VPN</li> <li>Маршрутизация</li> <li>Настройки MPLS</li> <li>Резервирование</li> <li>Управление QoS</li> <li>Мониторинг и управление</li> <li>Настройка SoftGRE контроллера туннелей</li> </ul>

Версия документа	Дата выпуска	Содержание изменений
Версия 1.20	03.02.2020	<p>Изменения в разделах:</p> <ul style="list-style-type: none"> <li>Структура системы команд</li> <li>Управление программным обеспечением и конфигурацией</li> <li>Конфигурирование и мониторинг туннелей</li> <li>Управление L2 функциями</li> <li>Управление Firewall</li> <li>Управление логированием и защитой от сетевых атак</li> <li>Управление NAT</li> <li>Настройки IPsec VPN</li> <li>Настройка протокола BFD</li> <li>Настройки MPLS</li> <li>Резервирование</li> <li>Управление QoS</li> <li>Мониторинг и управление</li> <li>Настройка Wi-Fi контроллера туннелей</li> </ul>
Версия 1.19	04.02.2020	<p>Добавлены разделы:</p> <ul style="list-style-type: none"> <li>Настройка mDNS-reflector</li> <li>Настройки MPLS</li> </ul> <p>Изменения в разделах:</p> <ul style="list-style-type: none"> <li>Структура системы команд</li> <li>Команды пользовательского интерфейса</li> <li>Настройка AAA</li> <li>Конфигурирование и мониторинг интерфейсов</li> <li>Конфигурирование и мониторинг туннелей</li> <li>Управление L2-функциями</li> <li>Настройка VRF</li> <li>Управление профилями</li> <li>Управление Firewall</li> <li>Настройки IPsec VPN</li> <li>Управление VPN. Настройки удаленного доступа</li> <li>Маршрутизация</li> <li>Управление QoS</li> <li>Управление sFlow</li> <li>Мониторинг и управление</li> <li>Настройка контроля абонентов (BRAS)</li> <li>Настройка VoIP</li> </ul>
Версия 1.18	08.11.2019	Синхронизация с версией ПО 1.8.2

Версия документа	Дата выпуска	Содержание изменений
Версия 1.17	02.09.2019	<p>Добавлены разделы:</p> <ul style="list-style-type: none"> <li>- 36 Настройка системы предотвращения вторжений (IPS/IDS)</li> </ul> <p>Изменения в разделах:</p> <ul style="list-style-type: none"> <li>- 1.4 Используемые сокращения</li> <li>- 3.2 Конфигурирование маршрутизатора</li> <li>- 5 Управление программным обеспечением и конфигурацией</li> <li>- 9 Настройка AAA</li> <li>- 10 Конфигурирование и мониторинг интерфейсов</li> <li>- 11 Конфигурирование и мониторинг туннелей</li> <li>- 12 Управление L2 функциями</li> <li>- 13 Работа с адресными таблицами</li> <li>- 14 Настройка VRF</li> <li>- 17 Управление профилями</li> <li>- 19 Управление firewall</li> <li>- 24 Управление VPN. Настройки удаленного доступа</li> <li>- 20 Управление фильтрацией</li> <li>- 25 Маршрутизация</li> <li>- 27 Управление QOS</li> <li>- 30 Мониторинг и управление</li> <li>- 35 Настройка VOIP</li> </ul>
Версия 1.16	15.08.2019	<p>Изменения в разделах:</p> <ul style="list-style-type: none"> <li>- 1.4 Используемые сокращения</li> <li>- 3.2 Конфигурирование маршрутизатора</li> <li>- 4 Команды пользовательского интерфейса</li> <li>- 5 Управление программным обеспечением и конфигурацией</li> <li>- 9 Настройка AAA</li> <li>- 10 Конфигурирование и мониторинг интерфейсов</li> <li>- 11 Конфигурирование и мониторинг туннелей</li> <li>- 23 Настройки IPsec VPN</li> <li>- 24 Управление VPN. Настройки удаленного доступа</li> <li>- 25 Маршрутизация</li> <li>- 27 Управление QOS</li> <li>- 30 Мониторинг и управление</li> <li>- 32 Настройка SLA</li> <li>- 33 Настройка контроля абонентов (BRAS)</li> <li>- 34 Настройка Wi-Fi контроллера туннелей</li> </ul>

Версия документа	Дата выпуска	Содержание изменений
Версия 1.15	05.08.2019	<p>Добавлены устройства ESR-20, ESR-21, ESR-1500, ESR-1510</p> <p>Добавлены разделы:</p> <ul style="list-style-type: none"> <li>- 10.5 Последовательные интерфейсы</li> <li>- 20 Управление фильтрацией</li> <li>- 32 Настройка SLA</li> </ul> <p>Изменения в разделах:</p> <ul style="list-style-type: none"> <li>- 3 Структура системы команд</li> <li>- 4 Команды пользовательского интерфейса</li> <li>- 5 Управление программным обеспечением и конфигурацией</li> <li>- 7 Настройка общесистемных параметров</li> <li>- 8 Управление системными часами</li> <li>- 9 Настройка AAA</li> <li>- 10 Конфигурирование и мониторинг интерфейсов</li> <li>- 11 Конфигурирование и мониторинг туннелей</li> <li>- 12 Управление L2 функциями</li> <li>- 13 Работа с адресными таблицами</li> <li>- 14 Настройка VRF</li> <li>- 15 Настройка IP-адресации</li> <li>- 16 Настройка IPV6 адресации</li> <li>- 17 Управления профилями</li> <li>- 18 Управление списками контроля доступа (ACL)</li> <li>- 19 Управление Firewall</li> <li>- 21 Управление логированием и защитой от сетевых атак</li> <li>- 22 Управление NAT</li> <li>- 23 Настройки IPsec VPN</li> <li>- 24 Управление VPN. Настройки удаленного доступа</li> <li>- 25 Маршрутизация</li> <li>- 26 Резервирование</li> <li>- 27 Управление QOS</li> <li>- 28 Управление Netflow</li> <li>- 29 Управление SFLOW</li> <li>- 30 Мониторинг и управление</li> <li>- 31 Настройка DHCP</li> </ul>
Версия 1.14	31.01.2019	<p>Изменения в разделах:</p> <ul style="list-style-type: none"> <li>- 11 Конфигурирование и мониторинг туннелей</li> </ul>
Версия 1.12	11.11.2018	Синхронизация с ПО 1.4.1
Версия 1.11	03.05.2018	Синхронизация с ПО 1.4.0

Версия документа	Дата выпуска	Содержание изменений
Версия 1.10	03.05.2017	<p>Изменения в разделах:</p> <ul style="list-style-type: none"> <li>- 3.2 Конфигурирование маршрутизатора</li> <li>- 4 Команды пользовательского интерфейса</li> <li>- 6 Настройка общесистемных параметров</li> <li>- 7 Управление системными часами</li> <li>- 8 Настройка AAA</li> <li>- 9.1 Управление интерфейсами</li> <li>- 11 Управление туннелями</li> <li>- 13 Управление адресными таблицами</li> <li>- 19 Управление Firewall</li> <li>- 21 Управление VPN. Настройка IPsec</li> <li>- 23.8 Настройка протокола BGP</li> <li>- 24.5 Настройка MultiWAN</li> <li>- 25 Управление QoS</li> <li>- 31 Настройка контроля абонентов (BRAS)</li> </ul>
Версия 1.9	16.12.2016	<p>Добавлены разделы:</p> <ul style="list-style-type: none"> <li>- 23.5 Настройка объектов отслеживания событий</li> <li>- 23.6 Настройка протокола BFD</li> <li>- 23.8 Настройка протокола BGP</li> <li>- 31 Настройка контроля абонентов (BRAS)</li> </ul> <p>Изменения в разделах:</p> <ul style="list-style-type: none"> <li>- 3.2 Конфигурирование маршрутизатора</li> <li>- 3.3 Типы и порядок именования интерфейсов маршрутизатора</li> <li>- 3.4 Типы и порядок именования туннелей маршрутизатора</li> <li>- 4 Команды пользовательского интерфейса</li> <li>- 6 Настройка общесистемных параметров</li> <li>- 8 Настройка AAA</li> <li>- 9 Настройка и мониторинг интерфейсов</li> <li>- 11 Управление туннелями</li> <li>- 26 Управление Netflow</li> <li>- 32 Настройка Wi-Fi контроллера туннелей</li> </ul>



Версия документа	Дата выпуска	Содержание изменений
Версия 1.8	22.07.2016	<p>Изменения в разделах:</p> <ul style="list-style-type: none"> <li>- 7 Управление системными часами</li> <li>- 8 Настройка AAA</li> <li>- 9.1 Ethernet-интерфейсы</li> <li>- 11 Управление туннелями</li> <li>- 24 Управление VRRP</li> <li>- 20 Управление NAT</li> <li>- 14 Настройка VRF</li> <li>- 12.1 Управление L2 маршрутизацией</li> <li>- 24.3.2 Настройка резервирования Firewall</li> </ul>
Версия 1.7	28.01.2015	<p>Добавлены разделы:</p> <ul style="list-style-type: none"> <li>- 22.2 OPENVPN</li> <li>- 30 Настройка wiSLA (система мониторинга качества услуг)</li> </ul> <p>Изменения в разделах:</p> <ul style="list-style-type: none"> <li>- 5.8 Управление программным обеспечением</li> <li>- 5.14 Управление программным обеспечением</li> <li>- 6 Настройка общесистемных параметров</li> <li>- 8 Настройка AAA</li> <li>- 9.1 Ethernet-интерфейсы</li> <li>- 11.13 Управление туннелями</li> <li>- 18.12 Управление списками контроля доступ</li> <li>- 22.1 L2TP/PPTP</li> <li>- 22.3 Общие команды настройки удаленного доступа</li> <li>- 23.7 Настройка протокола BGP</li> <li>- 20.8 Управление NAT</li> <li>- 24.1 Управление VRRP</li> <li>- 11 Управление туннелями</li> <li>- 14.4 Настройка VRF</li> <li>- 20.24 Управление NAT</li> <li>- 24.4 MultiWAN</li> <li>- 26 Управление QoS</li> <li>- 28.1 Настройка SNMP</li> <li>- 23.1 Общие настройки маршрутизации</li> <li>- 23.5 Настройка статических маршрутов IPv4/IPv6</li> <li>- 23.8 Настройка протокола OSPF</li> <li>- 23.9 Настройка протокола OSPFv3</li> <li>- 31 Настройка Wi-Fi контроллера туннелей</li> <li>- 25 Управление QOS</li> </ul>

Версия документа	Дата выпуска	Содержание изменений
Версия 1.6	17.08.2015	<p>Добавлены разделы:</p> <ul style="list-style-type: none"> <li>- 26 Управление Netflow</li> <li>- 27 Управление Sflow</li> </ul> <p>Изменения в разделах:</p> <ul style="list-style-type: none"> <li>- 2 Правила пользования командной строкой</li> <li>- 3.2 Конфигурирование маршрутизатора</li> <li>- 5 Управление программным обеспечением</li> <li>- 7 Управление системными часами</li> <li>- 8 Настройка AAA</li> <li>- 9.1 Ethernet-интерфейсы</li> <li>- 12.1 Управление L2 маршрутизацией</li> <li>- 12.3 Настройка и мониторинг VLAN</li> <li>- 13 Работа с адресными таблицами</li> <li>- 19 Управление Firewall</li> <li>- 20 Управление NAT</li> <li>- 21.2 Управление VPN. Настройка IPsec</li> <li>- 22 Управление VPN. Настройки удаленного доступа</li> <li>- 23. 1 Общие настройки маршрутизации</li> <li>- 23.2 Общие команды анонсирования и приема маршрутов</li> <li>- 23.3 Маршрутизация на основе политик (PBR)</li> <li>- 23.7 Настройка протокола BGP</li> <li>- 23.8 Настройка протокола RIP</li> <li>- 23.9 Настройка протокола OSPF</li> <li>- 28.1 Настройка SNMP</li> <li>- 28.3 Настройка доступа SSH, Telnet</li> <li>- 29 Настройка DHCP</li> </ul>

Версия документа	Дата выпуска	Содержание изменений
Версия 1.5	22.06.2015	<p>Добавлены разделы:</p> <ul style="list-style-type: none"> <li>- 9 Настройка AAA</li> <li>- 11 Управление IPv6 DHCP-клиентом</li> <li>- 13 Управление IPv6 DHCP-relay</li> <li>- 15 Управление IPv6 DHCP-сервером</li> <li>- 16.2 TDM(E1)</li> <li>- 17.2 MLPPP</li> <li>- 21 Настройка VRF</li> <li>- 23 Настройка IPv6 адресации</li> <li>- 25 Управление списками доступа (ACL)</li> <li>- 35 Маршрутизация на основе политик (PBR)</li> <li>- 38 Настройка статических IPv6 маршрутов</li> <li>- 42 Настройка протокола OSPFv3</li> <li>- 44 Настройка резервирования</li> </ul> <p>Изменения в разделах:</p> <ul style="list-style-type: none"> <li>- 4 Команды пользовательского интерфейса</li> <li>- 5 Управление программным обеспечением</li> <li>- 14 Настройка и мониторинг DHCP-сервера</li> <li>- 16.1 Ethernet интерфейсы</li> <li>- 20 Работа с адресными таблицами</li> <li>- 24 Управление профилями IP-адресов и портов</li> <li>- 26 Управление Firewall</li> <li>- 34 Общие команды анонсирования и приема маршрутов</li> <li>- 37 Настройка статических маршрутов</li> <li>- 39 Настройка протокола BGP</li> <li>- 40 Настройка протокола RIP</li> <li>- 41 Настройка протокола OSPF</li> <li>- 47 Настройка QoS</li> <li>- 50 Настройка SNMP</li> </ul>

Версия документа	Дата выпуска	Содержание изменений
Версия 1.4	11.03.2015	<p>Добавлены разделы:</p> <ul style="list-style-type: none"> <li>- 11 Управление DHCP агентом</li> <li>- 27 Общие настройки маршрутизации</li> <li>- 29 Настройка связок ключей</li> <li>- 34 Управление VRRP</li> <li>- 35 Управление DualHoming</li> <li>- 36 Настройка MultiWAN</li> <li>- 37 Управление QOS</li> <li>- 38 Настройка зеркалирования</li> <li>- 39 Настройка Wi-Fi контроллера</li> <li>- 40 Настройка SNMP</li> </ul> <p>Изменения в разделах:</p> <ul style="list-style-type: none"> <li>- 1.4 Используемые сокращения</li> <li>- 3.2 Конфигурирование маршрутизатора</li> <li>- 3.3 Типы и порядок именования интерфейсов маршрутизатора</li> <li>- 3.4 Типы и порядок именования туннелей маршрутизатора</li> <li>- 5 Управление программным обеспечением и конфигурацией</li> <li>- 6 Настройка общесистемных параметров</li> <li>- 7 Настройка доступа</li> <li>- 12 Настройка и мониторинг DHCP-сервера</li> <li>- 13 Настройка и мониторинг интерфейсов</li> <li>- 14 Управление Группами агрегации каналов – Link Agregation Group (LAG)</li> <li>- 20 Управление Firewall</li> <li>- 21 Управление NAT</li> <li>- 22 Управление VPN. Настройки IKE</li> <li>- 23 Управление VPN. Настройки IPsec</li> <li>- 25 Управление туннелями</li> <li>- 26 Управление Spanning Tree</li> <li>- 30 Настройка статических маршрутов</li> <li>- 31 Настройка протокола BGP</li> <li>- 32 Настройка протокола RIP</li> <li>- 33 Настройка протокола OSPF</li> <li>- 41 Управление Syslog</li> </ul>

Версия документа	Дата выпуска	Содержание изменений
Версия 1.3	25.11.2014	<p>Добавлены разделы:</p> <ul style="list-style-type: none"> <li>- 7 Настройка доступа</li> <li>- 13 Управление Группой агрегации каналов – Link Agregation Group (LAG)</li> <li>- 14 Настройка и мониторинг VLAN</li> <li>- 15 Управление bridge</li> <li>- 16 Работа с адресными таблицами</li> <li>- 24 Управление туннелями</li> <li>- 25 Управление STP, RSTP, MSTP</li> <li>- 26.2 Общие команды анонсирования маршрутов</li> <li>- 26.3 Настройка протокола BGP</li> <li>- 26.4 Настройка протокола RIP</li> <li>- 26.5 Настройка протокола OSPF</li> <li>- 27 Управление SYSLOG</li> </ul> <p>Изменения в разделах:</p> <ul style="list-style-type: none"> <li>- 3.2 Конфигурирование маршрутизатора</li> <li>- 3.3 Типы и порядок именования интерфейсов маршрутизатора</li> <li>- 4 Команды пользовательского интерфейса</li> <li>- 5 Управление программным обеспечением и конфигурацией</li> <li>- 6 Настройка общесистемных параметров</li> <li>- 12 Настройка и мониторинг интерфейсов</li> <li>- 18 Управление профилями IP-адресов и портов</li> <li>- 19 Управление Firewall</li> <li>- 20 Управление NAT</li> <li>- 22 Управление VPN. Настройки IPsec</li> <li>- 23 Управление VPN. Настройки удаленного доступа</li> </ul>

Версия документа	Дата выпуска	Содержание изменений
Версия 1.2	26.06.2014	<p>Синхронизация с версией ПО 1.0.2.</p> <p>Добавлены разделы:</p> <ul style="list-style-type: none"> <li>- 5 Настройка общесистемных параметров</li> <li>- 9 Управление системными часами</li> <li>- 18 Управление VPN. Настройки удаленного доступа</li> </ul> <p>Изменения в разделах:</p> <ul style="list-style-type: none"> <li>- 3.2 Конфигурирование маршрутизатора</li> <li>- 4 Команды пользовательского интерфейса</li> <li>- 6 Управление программным обеспечением и конфигурацией</li> <li>- 7 Настройка статических маршрутов</li> <li>- 10 Управление DHCP-клиентом</li> <li>- 12 Настройка и мониторинг интерфейсов</li> <li>- 14 Управление Firewall</li> <li>- 16 Управление VPN. Настройка IKE</li> <li>- 17 Управление VPN. Настройка IPsec</li> </ul>
Версия 1.1	29.04.2014	<p>Синхронизация с версией ПО 1.0.1.</p> <p>Раздел «Отладка работы устройства» перенесен в отдельный документ.</p>
Версия 1.0	20.03.2014	Первая публикация
<b>Версия программного обеспечения</b>	<b>1.30</b>	

### 3 Правила пользования командной строкой

Для упрощения использования командной строки интерфейс поддерживает функцию автоматического дополнения команд. Эта функция активируется при неполно набранной команде и вводе символа табуляции <Tab>.

Другая функция, помогающая пользоваться командной строкой – контекстная подсказка. На любом этапе ввода команды можно получить подсказку о следующих элементах команды путем ввода вопросительного знака <?>.

Для упрощения команд всей системе команд придана иерархическая структура. Для перехода между уровнями иерархии предназначены специальные команды перехода. Это позволяет использовать менее объемные команды на каждом из уровней. Для обозначения текущего уровня, на котором находится пользователь, динамически изменяется строка приглашения системы.

#### Пример:

```
esr> enable Включение 15 уровня привилегий
```

```
esr# configure Переход в режим конфигурирования устройства
```

```
esr(config)#
esr(config)# exit возврат на уровень выше
esr#
```

Для удобства использования командной строки реализована поддержка горячих клавиш, перечисленных в таблице 1.

Таблица 1 – Описание горячих клавиш командной строки CLI

Сочетание клавиши	Описание
Ctrl+D	Во вложенном командном режиме – выход в предыдущий командный режим (команда exit), в корневом командном режиме – выход из CLI (команда logout)
Ctrl+Z	Выход в корневой командный режим (команда top)
Ctrl+A	Переход в начало строки
Ctrl+E	Переход в конец строки
Ctrl+U	Удаление символов слева от курсора
Ctrl+K	Удаление символов справа от курсора
Ctrl+C	Очистка строки, а также обрыв выполнения команды
Ctrl+W	Удаление слова слева от курсора
Ctrl+B	Переход курсора на одну позицию назад
Ctrl+F	Переход курсора на одну позицию вперед
Ctrl+L	Очистка экрана

Для удобства чтения добавлен постраничный вывод большой по объему информации.

### Пример:

```
esr# show running-config
syslog max-files 3
syslog file-size 512
syslog file esr info
syslog console info
interface gigabitethernet 1/0/1
  ip address 10.100.14.1/24
exit
interface gigabitethernet 1/0/1.101
exit
interface gigabitethernet 1/0/2
  ip address 192.168.1.1/24
  ip address 10.100.100.2/24 secondary
exit
interface gigabitethernet 1/0/2.150
  ip address 10.150.150.2/24
exit
interface gigabitethernet 1/0/2.151
  ip address 10.151.151.15/24
exit
interface gigabitethernet 1/0/3
  ip address dhcp enable
exit
interface gigabitethernet 1/0/5.55
More? Enter - next line; Space - next page; Q - quit; R - show the rest.
```

Для отключения постраничного вывода в текущей сессии необходимо ввести команду:

```
esr# terminal datadump
```

Интерфейс командной строки обеспечивает авторизацию пользователей и ограничивает доступ к командам на основании уровня привилегий, заданного администратором.

В системе может быть создано необходимое количество пользователей. Необходимый уровень привилегий задаётся индивидуально для каждого из них.

- ✓ В заводской конфигурации в системе создан один пользователь с именем **admin** и паролем **password**.



Для обеспечения безопасности командного интерфейса команды распределены между 1, 10 и 15 уровнем привилегий:

- 1 уровень – доступен только мониторинг устройства;
- 10 уровень – доступно конфигурирование устройства, кроме создания пользователей, перезагрузки устройства, загрузки ПО;
- 15 уровень – нет ограничений.

Получение 15 уровня привилегий:

```
(esr)> enable
(esr)#
```

Возвращение на первоначальный уровень привилегий:

```
(esr)# disable
(esr)>
```

Система позволяет нескольким пользователям одновременно подключаться к устройству.

При вводе в качестве параметров строк, содержащих пробелы, вводимые данные можно заключить в кавычки или апострофы:

```
esr(config)# interface gigabitethernet 1/0/1
esr(config-if-gi)# description "[WAN] ISP-1 link"
esr(config-if-gi)# exit
esr(config)# interface gigabitethernet 1/0/2
esr(config-if-gi)# description '[WAN] ISP-2 link'
esr(config-if-gi)#
```

Основное отличие между кавычками и апострофами заключается в том, что при использовании кавычек часть спецсимволов внутри для корректного ввода необходимо экранировать, а текст между апострофами интерпретируется "как есть":

```
esr(config)# banner login "first line\nsecond line"
esr(config)# banner exec 'first line\nsecond line'
esr(config)#
```

Для уменьшения объема отображаемых данных в ответ на запросы пользователя и облегчения поиска необходимой информации можно воспользоваться фильтрацией. Для фильтрации информации требуется добавить в конец командной строки символ «|» и использовать одну из опций фильтрации:

- begin – выводить все после строки, содержащей заданный шаблон;
- count – выводить только количество строк, отображаемых в выводе команды (без вывода информации, отображаемой самой командой);
- counter – добавляет к выводимой информации номера строк;
- include – выводить все строки, содержащие заданный шаблон;
- exclude – выводить все строки, не содержащие заданный шаблон;
- until – выводить все до строки, содержащей заданный шаблон.

Шаблон поиска может быть задан регулярным выражением и содержать:

– Перечень символов. Можно определить перечень, заключив символы в квадратные скобки «[]». Соответствие будет проверяться по символам, перечисленным в перечне. Если первый символ перечня «^», то соответствие будет проверяться по любому символу, не входящему в перечень. Примеры:

- [-az] – 'a', 'z' и '-';
- [a-z] – все латинские буквы от 'a' до 'z'.

– Специальные символы и операторы:

- ^ – начало строки;
- \$ – конец строки;
- . – любой символ в строке;
- \* – ноль или более раз;
- | – или.

Вывод команды «show running-config syslog» без параметров:

```
esr# show running-config syslog
syslog max-files 3
syslog file-size 512
syslog file default info
```

Вывод команды «show running-config syslog» с параметром «begin»:

```
esr# show running-config syslog | begin file-size
syslog file-size 512
syslog file default info
```

Вывод команды «show running-config syslog» с параметром «include»:

```
esr# show running-config syslog | include file-size
syslog file-size 512
```

Вывод команды «show running-config syslog» с параметром «exclude»:

```
esr# show running-config syslog | exclude file-size
syslog max-files 3
syslog file default info
```

Примеры использования регулярных выражений:

```
esr# show interfaces status | include "^te.*"
te1/0/1      Up      Down    1500    a8:f9:4b:aa:05:d9
te1/0/2      Up      Down    1500    a8:f9:4b:aa:05:da
esr# show interfaces status | include "^gi1/0/1[2568]"
gi1/0/12     Up      Down    1500    a8:f9:4b:aa:05:cc
gi1/0/15     Up      Down    1500    a8:f9:4b:aa:05:cf
gi1/0/16     Up      Down    1500    a8:f9:4b:aa:05:d0
gi1/0/18     Up      Down    1500    a8:f9:4b:aa:05:d2
esr# show interfaces status | include "[^tgI -]"
bridge 1     Up      Up       1500    a8:f9:4b:aa:05:c0
bridge 2     Up      Up       1500    a8:f9:4b:aa:05:c0
esr# show interfaces status | include "(/3|/5)"
gi1/0/3      Up      Down    1500    a8:f9:4b:ab:0e:23
gi1/0/5      Up      Down    1500    a8:f9:4b:ab:0e:25
```

Для вывода информации, запрошенной командой show, в файл требуется добавить в конец командной строки соответствующий символ, путь и имя для создания или дополнения текстового файла.

Символы для вывода информации в файл:

- > – записать вывод команды в файл;
- >> – дописать вывод команды в конец файла.

Возможные пути для сохранения текстовых файлов вывода команд:

- flash:data/<FILE> – файл будет создан или дополнен в энергонезависимой памяти маршрутизатора в разделе flash:data;
- usb://<USB-NAME>:<FILE> – файл будет создан или дополнен на USB-носителе с именем <USB-NAME>;
- mmc://<MMC-NAME>:<FILE> – файл будет создан или дополнен на USB-носителе с именем <MMC-NAME>.

## 4 Структура системы команд

- Глобальный режим
- Конфигурирование маршрутизатора
- Типы и порядок именования интерфейсов маршрутизатора
- Типы и порядок именования туннелей маршрутизатора

Система команд интерфейса командной строки маршрутизатора серии ESR и контроллера беспроводного доступа WLC разделена на иерархические уровни (разделы).

### Глобальный режим

Верхний уровень иерархии команд приведен в таблице 2.

Таблица 2 – Иерархия командных режимов (верхний уровень)

Уровень	Команда входа	Вид строки подсказки	Команда выхода
Корневой режим ( <b>ROOT</b> )		esr> esr#	exit end
Режим конфигурирования ( <b>CONFIG</b> )	configure	esr(config)#	
Режим отладки работы устройства ( <b>DEBUG</b> )	debug	esr(debug)#	

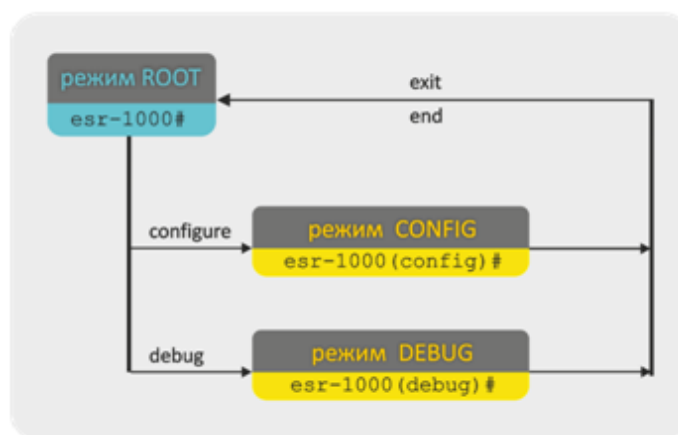


Рисунок 1 – Верхний уровень иерархии режимов команд

В корневом командном режиме (ROOT) осуществляется:

– работа с файлами конфигурации:

- применение;
- подтверждение;
- сброс;
- сохранение;
- отмена непримененных изменений;
- возврат к подтвержденной конфигурации.

– перезагрузка маршрутизатора;

– мониторинг работы и просмотр текущей конфигурации устройства.

Из корневого режима (ROOT) осуществляется переход к следующим разделам:

– режим конфигурирования устройства (CONFIG);

– режим отладки работы устройства (DEBUG).

## Конфигурирование маршрутизатора

Конфигурирование маршрутизатора серии ESR и контроллера беспроводного доступа WLC выполняется в режиме **CONFIG**. Данный режим доступен из корневого режима (ROOT). Переход в режим конфигурирования осуществляется только в привилегированном режиме.

Для перехода из корневого режима (ROOT) необходимо выполнить следующие команды:

```
esr> enable
esr# configure
esr(config)#
```

В режиме конфигурирования маршрутизатора серии ESR и контроллера беспроводного доступа WLC выполняется:

- управление системными часами;
- управление системным журналом;
- управление удаленным доступом;
- настройка QoS;
- настройка Spanning Tree;
- настройка VLAN;
- настройка статических маршрутов;
- настройка приоритетности протоколов маршрутизации;
- переход к режимам конфигурирования функций, описание режимов приведено в таблице 3.

Таблица 3 – Командные режимы для управления маршрутизатором и контроллером

Уровень	Команда входа <sup>1</sup>	Вид строки подсказки	Предыдущий уровень
Настройка глобальных параметров ( <b>CONFIG</b> )	<b>configure</b>	esr(config)#	ROOT
Настройка WLC ( <b>CONFIG-WLC</b> )	<b>wlc</b>	wlc(config-wlc)#	CONFIG
Настройка сервиса Airtune ( <b>CONFIG-AIRTUNE</b> )	<b>airtune</b>	wlc(config-airtune)#	CONFIG-WLC
Настройка профиля Airtune ( <b>CONFIG-AIRTUNE-PROFILE</b> )	<b>airtune-profile &lt;NAME&gt;</b>	wlc(config-airtune-profile)#	CONFIG-AIRTUNE

Уровень	Команда входа <sup>1</sup>	Вид строки подсказки	Предыдущий уровень
Настройка профиля определенной точки доступа <b>(CONFIG-WLC-AP)</b>	<b>ap &lt;MAC_AP&gt;</b>	wlc(config-wlc-ap)#	CONFIG-WLC
Настройка локации <b>(CONFIG-WLC-AP-LOCATION)</b>	<b>ap-location &lt;NAME&gt;</b>	wlc(config-wlc-ap)#	CONFIG-WLC
Настройка профиля общих настроек точек доступа <b>(CONFIG-WLC-AP-PROFILE)</b>	<b>ap-profile &lt;NAME&gt;</b>	wlc(config-wlc-ap-profile)#	CONFIG-WLC
Настройка трассировки <b>(CONFIG-WLC-AP-PROFILE-TRACE)</b>	<b>trace</b>	wlc(config-wlc-ap-profile-trace)#	CONFIG-WLC-AP-PROFILE
Настройка профиля радиопараметров в диапазоне 2.4 ГГц <b>(RADIO-2G-PROFILE)</b>	<b>radio-2g-profile &lt;NAME&gt;</b>	wlc(radio-2g-profile)#	CONFIG-WLC
Настройка профиля радиопараметров в диапазоне 5 ГГц <b>(RADIO-5G-PROFILE)</b>	<b>radio-5g-profile &lt;NAME&gt;</b>	wlc(radio-5g-profile)#	CONFIG-WLC
Настройка диапазона адресов <b>(CONFIG-WLC-IP-POOL)</b>	<b>ip-pool &lt;NAME&gt;</b>	wlc(config-wlc-ip-pool)#	CONFIG-WLC
Настройка профиля портала <b>(CONFIG-WLC-PORTAL-PROFILE)</b>	<b>portal-profile &lt;NAME&gt;</b>	wlc(config-wlc-portal-profile)#	CONFIG-WLC
Настройка профиля RADIUS для авторизации пользователей <b>(CONFIG-WLC-RADIUS-PROFILE)</b>	<b>radius-profile &lt;NAME&gt;</b>	wlc(config-wlc-radius-profile)#	CONFIG-WLC

Уровень	Команда входа <sup>1</sup>	Вид строки подсказки	Предыдущий уровень
Настройка сервис-активатора <b>(CONFIG-WLC-SERVICE-ACTIVATOR)</b>	<b>service-activator</b>	wlc(config-wlc-service-activator)#	CONFIG-WLC
Настройка SSID <b>(CONFIG-WLC-SSID-PROFILE)</b>	<b>ssid-profile &lt;NAME&gt;</b>	wlc(config-wlc-ssid-profile)#	CONFIG-WLC
Настройка расписания обновления ПО для точек доступа <b>(CONFIG-WLC-UPDATE-MANAGER)</b>	<b>update-manager</b>	wlc(config-wlc-update-manager)#	CONFIG-WLC
Настройка локального RADIUS-сервера <b>(CONFIG-RADIUS)</b>	<b>radius-server local</b>	wlc(config-radius)#	CONFIG-WLC
Настройка NAS <b>(CONFIG-RADIUS-NAS)</b>	<b>config-radius-nas</b>	wlc(config-radius-nas)#	CONFIG
Настройка виртуального RADIUS-сервера <b>(CONFIG-RADIUS-VSERVER)</b>	<b>config-radius-vserver</b>	wlc-30(config-radius-vserver)#	CONFIG-RADIUS
Настройка виртуального RADIUS-сервера как используемого сервера <b>(CONFIG-RADIUS-SERVER)</b>	<b>config-radius-server</b>	wlc(config-radius-server)#	CONFIG-RADIUS
Настройка профиля AAA <b>(CONFIG-AAA-RADIUS-PROFILE)</b>	<b>config-aaa-radius-profile</b>	wlc(config-aaa-radius-profile)#	CONFIG
Настройка GigabitEthernet-интерфейсов <b>(CONFIG-IF-GI)</b>	<b>interface gigabitethernet &lt;PORT&gt;</b>	esr(config-if-gi)#	CONFIG

Уровень	Команда входа <sup>1</sup>	Вид строки подсказки	Предыдущий уровень
Настройка TenggigabitEthernet-интерфейсов (CONFIG-IF-TE)	<b>interface tengigabitethernet &lt;PORT&gt;</b>	esr(config-if-te)#	CONFIG
Настройка Twentyfivegigabitethernet-интерфейсов (CONFIG-IF-TWE)	<b>interface twentyfivegigabitethernet &lt;PORT&gt;</b>	esr(config-if-twe)	CONFIG
Настройка FourtygigabitEthernet-интерфейсов (CONFIG-IF-FO)	<b>interface fourtygigabitethernet &lt;PORT&gt;</b>	esr(config-if-fo)#	CONFIG
Настройка HundredgigabitEthernet-интерфейсов (CONFIG-IF-HU)	<b>interface hundredgigabitethernet &lt;PORT&gt;</b>	esr(config-if-hu)#	CONFIG
Настройка OOB-интерфейса (CONFIG-IF-OOB)	<b>interface oob &lt;PORT&gt;</b>	esr(config-if-oob)#	CONFIG
Настройка саб-интерфейсов (CONFIG-IF-SUB)	<b>interface gigabitethernet &lt;PORT&gt;.&lt;VLAN&gt;</b> или <b>interface tengigabitethernet &lt;PORT&gt;.&lt;VLAN&gt;</b> или <b>interface port-channel &lt;CH&gt;.&lt;VLAN&gt;</b>	esr(config-if-sub)#	CONFIG
Настройка саб-интерфейсов (CONFIG-IF-QINQ)	<b>interface gigabitethernet &lt;PORT&gt;.&lt;VLAN&gt;.&lt;VLAN&gt;</b> или <b>interface tengigabitethernet &lt;PORT&gt;.&lt;VLAN&gt;.&lt;VLAN&gt;</b> или <b>interface port-channel &lt;CH&gt;.&lt;VLAN&gt;.&lt;VLAN&gt;</b>	esr(config-if-qinq)#	CONFIG
Настройка последовательных интерфейсов <sup>2</sup> (CONFIG-SERIAL)	<b>interface serial &lt;PORT&gt;</b>	esr(config-serial)#	CONFIG
Настройка виртуальных интерфейсов (CONFIG-LOOPBACK)	<b>interface loopback &lt;PORT&gt;</b>	esr(config-loopback)#	CONFIG
Настройка E1-интерфейсов (CONFIG-IF-E1)	<b>interface e1 &lt;PORT&gt;</b>	esr(config-if-e1)#	CONFIG



Уровень	Команда входа <sup>1</sup>	Вид строки подсказки	Предыдущий уровень
Настройка группы агрегации E1-каналов <b>(CONFIG-IF-MULTILINK)</b>	<b>interface multilink &lt;PORT&gt;</b>	esr(config-if-multilink)#	CONFIG
Настройка FXO-портов <b>(CONFIG-VOICE-PORT-FXO)</b>	<b>interface voice-port &lt;NUM&gt;</b>	esr(config-voice-port-fxo)#	CONFIG
Настройка FXS-портов <b>(CONFIG-VOICE-PORT-FXS)</b>	<b>interface voice-port &lt;NUM&gt;</b>	esr(config-voice-port-fxs)#	CONFIG
Настройка виртуальных туннельных интерфейсов VTI <b>(CONFIG-VTI)</b>	<b>tunnel vti &lt;VTI&gt;</b>	esr(config-vti)#	CONFIG
Настройка последовательного интерфейса <sup>2</sup> <b>(CONFIG-LINE-AUX)</b>	<b>line aux [ &lt;UNIT&gt;/&lt;SLOT&gt;/&lt;PORT&gt; ]</b>	esr(config-line-aux)#	CONFIG
Настройка L2TPv3-туннелей <b>(CONFIG-L2TPV3)</b>	<b>tunnel l2tpv3 &lt;L2TPV3&gt;</b>	esr(config-l2tpv3)#	CONFIG
Настройка GRE-туннелей <b>(CONFIG-GRE)</b>	<b>tunnel gre &lt;GRE&gt;</b>	esr(config-gre)#	CONFIG
Настройка SoftGRE-туннелей <b>(CONFIG-SOFTGRE)</b>	<b>tunnel softgre &lt;SOFTGRE&gt;</b>	esr(config-softgre)#	CONFIG
Настройка логических туннелей <b>(CONFIG-LT)</b>	<b>tunnel lt &lt;LT&gt;</b>	esr(config-lt)#	CONFIG
Настройка саб-интерфейса на L2-туннеле <b>(CONFIG-SUBTUNNEL)</b>	<b>tunnel softgre &lt;GRE&gt;.&lt;VLAN&gt;</b>	esr(config-subtunnel)#	CONFIG

Уровень	Команда входа <sup>1</sup>	Вид строки подсказки	Предыдущий уровень
Настройка IPv4-over-IPv4 туннелей <b>(CONFIG-IP4IP4)</b>	<b>tunnel ip4ip4 &lt;IP4IP4&gt;</b>	esr(config-ip4ip4)#	CONFIG
Настройка WireGuard туннелей <b>(CONFIG-WIREGUARD)</b>	<b>tunnel wireguard &lt;WIREGUARD&gt;</b>	esr(config-wireguard)#	CONFIG
Настройка пира для WireGuard-туннеля <b>(CONFIG-WIREGUARD-TUNNEL-PEER)</b>	<b>peer &lt;PEER&gt;</b>	esr(config-wireguard-tunnel-peer)#	CONFIG-WIREGUARD
Настройка сетевых мостов <b>(CONFIG-BRIDGE)</b>	<b>bridge &lt;BRIDGE&gt;</b>	esr(config-bridge)#	CONFIG
Настройка VLAN <b>(CONFIG-VLAN)</b>	<b>vlan &lt;VLAN&gt;</b>	esr(config-vlan)#	CONFIG
Настройка сетевой политики для определения VLAN по LLDPDU-сообщениям клиента <b>(CONFIG-NET-POLICY)</b>	<b>network-policy &lt;NAME&gt;</b>	esr(config-net-policy)	CONFIG
Настройка пула адресов DHCP-сервера <b>(CONFIG-DHCP-SERVER)</b>	<b>ip dhcp-server pool &lt;NAME&gt;</b>	esr(config-dhcp-server)#	CONFIG
Настройка DHCP опции 60 <b>(CONFIG-DHCP-VENDOR-ID)</b>	<b>ip dhcp-server vendor-class-id &lt;NAME&gt;</b>	esr(config-dhcp-vendor-id)#	CONFIG
Настройка пула адресов DHCP-сервера <b>(CONFIG-IPV6-DHCP-SERVER)</b>	<b>ipv6 dhcp-server pool &lt;NAME&gt;</b>	esr(config-ipv6-dhcp-server)#	CONFIG
Настройка DHCP опции 60 <b>(CONFIG-IPV6-DHCP-VENDOR-ID)</b>	<b>ipv6 dhcp-server vendor-class-id &lt;NAME&gt;</b>	esr(config-dhcp-vendor-id)#	CONFIG

Уровень	Команда входа <sup>1</sup>	Вид строки подсказки	Предыдущий уровень
Настройка резервирования DHCP-сервера <b>(CONFIG-DHCP-SERVER-FAILOVER)</b>	<b>ip dhcp-server failover [ vrf &lt;VRF&gt; ]</b>	esr(config-dhcp-server-failover)#	CONFIG
Настройка синхронизации крипто-файлов <b>(CONFIG-CRYPTO-SYNC)</b>	<b>crypto-sync</b>	esr(config-crypto-sync)#	CONFIG
Настройка профиля приложений <b>(CONFIG-OBJECT-GROUP-APPLICATION)</b>	<b>object-group application &lt;NAME&gt;</b>	esr(config-object-group-application)#	CONFIG
Настройка профиля IP-адресов <b>(CONFIG-OBJECT-GROUP-NETWORK)</b>	<b>object-group network &lt;NAME&gt;</b>	esr(config-object-group-network)#	CONFIG
Настройка профиля связок IP-адресов и TCP/UDP-портов <b>(CONFIG-OBJECT-GROUP-ADDRESS-PORT)</b>	<b>object-group address-port &lt;NAME&gt;</b>	esr(config-object-group-address-port)#	CONFIG
Настройка профиля TCP/UDP-портов <b>(CONFIG-OBJECT-GROUP-SERVICE)</b>	<b>object-group service &lt;NAME&gt;</b>	esr(config-object-group-service)#	CONFIG
Настройка профиля URL <b>(CONFIG-OBJECT-GROUP-URL)</b>	<b>object-group url &lt;NAME&gt;</b>	esr(config-object-group-url)#	CONFIG
Настройка поставщика категорий контентной фильтрации <b>(CONFIG-OBJECT-GROUP-CONTENT-FILTER)</b>	<b>object-group content-filter &lt;NAME&gt;</b>	esr(config-object-group-content-filter)#	CONFIG

Уровень	Команда входа <sup>1</sup>	Вид строки подсказки	Предыдущий уровень
Настройка профиля категорий контентной фильтрации от вендора Kaspersky Laboratory <b>(CONFIG-OBJECT-GROUP-CF-KASPERSKY)</b>	<b>vendor &lt;VENDOR&gt;</b>	esr(config-object-group-cf-kaspersky)#	CONFIG-OBJECT-GROUP-CONTENT-FILTER
Настройка профиля почтовых доменов и адресов почтовых ящиков <b>(CONFIG-OBJECT-GROUP-MAIL)</b>	<b>object-group email &lt;NAME&gt;</b>	esr(config-object-group-email)#	CONFIG
Настройка профиля XAUTH <b>(CONFIG-ACCESS-PROFILE)</b>	<b>access profile &lt;NAME&gt;</b>	esr(config-access-profile)#	CONFIG
Настройка пула IP-адресов <b>(CONFIG-POOL)</b>	<b>address-assignment pool &lt;NAME&gt;</b>	esr(config-pool)#	CONFIG
Настройка шлюза протокола IKE <b>(CONFIG-IKE-GATEWAY)</b>	<b>security ike gateway &lt;NAME&gt;</b>	esr(config-ike-gw)#	CONFIG
Настройка политики протокола IKE <b>(CONFIG-IKE-POLICY)</b>	<b>security ike policy &lt;NAME&gt;</b>	esr(config-ike-policy)#	CONFIG
Настройка профиля протокола IKE <b>(CONFIG-IKE-PROPOSAL)</b>	<b>security ike proposal &lt;NAME&gt;</b>	esr(config-ike-proposal)#	CONFIG
Настройка политики набора протоколов IPsec <b>(CONFIG-IPSEC-POLICY)</b>	<b>security ipsec policy &lt;NAME&gt;</b>	esr(config-ipsec-policy)#	CONFIG
Настройка профиля набора протоколов IPsec <b>(CONFIG-IPSEC-PROPOSAL)</b>	<b>security ipsec proposal &lt;NAME&gt;</b>	esr(config-ipsec-proposal)#	CONFIG

Уровень	Команда входа <sup>1</sup>	Вид строки подсказки	Предыдущий уровень
Настройка VPN на основе набора протоколов IPsec <b>(CONFIG-IPSEC-VPN)</b>	<b>security ipsec vpn &lt;NAME&gt;</b>	esr(config-ipsec-vpn)#	CONFIG
Настройка списка контроля доступа <b>(CONFIG-ACL)</b>	<b>ip access-list extended &lt;NAME&gt;</b>	esr(config-acl)#	CONFIG
Настройка правила для списка контроля доступа <b>(CONFIG-ACL-RULE)</b>	<b>rule &lt;ORDER&gt;</b>	esr(config-acl-rule)#	CONFIG-ACL
Настройка зоны безопасности <b>(CONFIG-ZONE)</b>	<b>security zone &lt;NAME&gt;</b>	esr(config-zone)#	CONFIG
Настройка группы правил для пар зон безопасности <b>(CONFIG-ZONE-PAIR)</b>	<b>security zone-pair &lt;FROM&gt; &lt;TO&gt;</b>	esr(config-zone-pair)#	CONFIG
Настройка правила для пары зон безопасности <b>(CONFIG-ZONE-PAIR-RULE)</b>	<b>rule &lt;ORDER&gt;</b>	esr(config-zone-rule)#	CONFIG-ZONE-PAIR
Настройка сервиса трансляции адресов получателя <b>(CONFIG-DNAT)</b>	<b>nat destination</b>	esr(config-dnat)#	CONFIG
Настройка пула IP-адресов и TCP/UDP-портов для DNAT <b>(CONFIG-DNAT-POOL)</b>	<b>pool &lt;NAME&gt;</b>	esr(config-dnat-pool)#	CONFIG-DNAT
Настройка группы правил для DNAT <b>(CONFIG-DNAT-RULESET)</b>	<b>ruleset &lt;NAME&gt;</b>	esr(config-dnat-ruleset)#	CONFIG-DNAT
Настройка правила для DNAT <b>(CONFIG-DNAT-RULE)</b>	<b>rule &lt;ORDER&gt;</b>	esr(config-dnat-rule)#	CONFIG-DNAT-RULESET

Уровень	Команда входа <sup>1</sup>	Вид строки подсказки	Предыдущий уровень
Настройка сервиса трансляции адресов отправителя <b>(CONFIG-SNAT)</b>	<b>nat source</b>	esr(config-snat)#	CONFIG
Настройка пула IP-адресов и TCP/UDP-портов для SNAT <b>(CONFIG-SNAT-POOL)</b>	<b>pool &lt;NAME&gt;</b>	esr(config-snat-pool)#	CONFIG-SNAT
Настройка группы правил для SNAT <b>(CONFIG-SNAT-RULESET)</b>	<b>ruleset &lt;NAME&gt;</b>	esr(config-snat-ruleset)#	CONFIG-SNAT
Настройка правила для SNAT <b>(CONFIG-SNAT-RULE)</b>	<b>rule &lt;ORDER&gt;</b>	esr(config-snat-rule)#	CONFIG-SNAT-RULESET
Настройка пользователей системы <b>(CONFIG-USER)</b>	<b>username &lt;NAME&gt;</b>	esr(config-user)#	CONFIG
Настройка локальной консоли <b>(CONFIG-LINE-CONSOLE)</b>	<b>line console</b>	esr(config-line-console)#	CONFIG
Настройка удаленной консоли <b>(CONFIG-LINE-TELNET)</b>	<b>line telnet</b>	esr(config-line-telnet)#	CONFIG
Настройка защищенной удаленной консоли <b>(CONFIG-LINE-SSH)</b>	<b>line ssh</b>	esr(config-line-ssh)#	CONFIG
Настройка TACACS-сервера <b>(CONFIG-TACACS-SERVER)</b>	<b>tacacs-server host &lt;ADDR&gt;</b>	esr(config-tacacs-server)#	CONFIG
Настройка RADIUS-сервера <b>(CONFIG-RADIUS-SERVER)</b>	<b>radius-server host &lt;ADDR&gt;</b>	esr(config-radius-server)#	CONFIG

Уровень	Команда входа <sup>1</sup>	Вид строки подсказки	Предыдущий уровень
Настройка профиля RADIUS-серверов <b>(CONFIG-RADIUS-SERVER-PROFILE)</b>	<b>aaa radius-profile &lt;NAME&gt;</b>	esr(config-aaa-radius-profile)#	CONFIG
Настройка DAS-сервера <b>(CONFIG-DAS-SERVER)</b>	<b>das-server &lt;NAME&gt;</b>	esr(config-das-server)#	CONFIG
Настройка профиля DAS-серверов <b>(CONFIG-DAS-SERVER-PROFILE)</b>	<b>aaa das-profile &lt;NAME&gt;</b>	esr(config-aaa-das-profile)#	CONFIG
Настройка LDAP-сервера <b>(CONFIG-LDAP-SERVER)</b>	<b>ldap-server host &lt;ADDR&gt;</b>	esr(config-ldap-server)#	CONFIG
Настройка сервера для удаленного доступа по протоколу PPTP <b>(CONFIG-PPTP-SERVER)</b>	<b>remote-access pptp &lt;NAME&gt;</b>	esr(config-pptp-server)#	CONFIG
Настройка пользователей для удаленного доступа по протоколу PPTP <b>(CONFIG-PPTP-USER)</b>	<b>username &lt;NAME&gt;</b>	esr(config-pptp-user)#	CONFIG-PPTP
Настройка сервера для удаленного доступа по протоколу L2TP <b>(CONFIG-L2TP-SERVER)</b>	<b>remote-access l2tp &lt;NAME&gt;</b>	esr(config-l2tp-server)#	CONFIG
Настройка пользователей для удаленного доступа по протоколу L2TP <b>(CONFIG-L2TP-USER)</b>	<b>username &lt;NAME&gt;</b>	esr(config-l2tp-user)#	CONFIG-L2TP-SERVER

Уровень	Команда входа <sup>1</sup>	Вид строки подсказки	Предыдущий уровень
Настройка сервера для удаленного доступа по протоколу OpenVPN <b>(CONFIG-OPENVPN-SERVER)</b>	<b>remote-access openvpn &lt;NAME&gt;</b>	esr(config-openvpn-server)#	CONFIG
Настройка пользователей для удаленного доступа по протоколу L2TP <b>(CONFIG-OPENVPN-USER)</b>	<b>username &lt;NAME&gt;</b>	esr(config-openvpn-user)#	CONFIG-OPENVPN-SERVER
Настройка сервера для удаленного доступа по протоколу WireGuard <b>(CONFIG-WIREGUARD-SERVER)</b>	<b>remote-access wireguard &lt;NAME&gt;</b>	esr(config-wireguard-server)#	CONFIG
Настройка пира для удаленного доступа по протоколу WireGuard <b>(CONFIG-WIREGUARD-SERVER-PEER)</b>	<b>peer &lt;PEER&gt;</b>	esr(config-wireguard-server-peer)#	CONFIG-WIREGUARD-SERVER
Настройка клиента удаленного доступа по протоколу PPTP <b>(CONFIG-PPTP)</b>	<b>tunnel pptp &lt;PPTP-ID&gt;</b>	esr(config-pptp)#	CONFIG
Настройка клиента удаленного доступа по протоколу PPPOE <b>(CONFIG-PPPOE)</b>	<b>tunnel pppoe &lt;PPPOE-ID&gt;</b>	esr(config-pppoe)#	CONFIG
Настройка клиента удаленного подключения по протоколу L2TP <b>(CONFIG-L2TP)</b>	<b>tunnel l2tp &lt;L2TP-ID&gt;</b>	esr(config-l2tp)#	CONFIG
Настройка клиента удаленного подключения по протоколу OpenVPN <b>(CONFIG-OPENVPN)</b>	<b>tunnel openvpn &lt;OPENVPN-ID&gt;</b>	esr(config-openvpn)#	CONFIG



Уровень	Команда входа <sup>1</sup>	Вид строки подсказки	Предыдущий уровень
Настройка SNMP-пользователя <b>(CONFIG-SNMP-USER)</b>	<b>snmp-server &lt;NAME&gt;</b>	esr(config-snmp-user)#	CONFIG
Настройка NTP-сервера или пира <b>(CONFIG-NTP)</b>	<b>ntp server &lt;ADDR&gt;</b> <b>ntp peer &lt;ADDR&gt;</b>	esr(config-ntp)#	CONFIG
Настройка BGP-процесса <b>(CONFIG-BGP)</b>	<b>router bgp &lt;AS&gt;</b>	esr(config-bgp)#	CONFIG
Настройка BGP-процесса в VRF <b>(CONFIG-BGP-VRF)</b>	<b>vrf &lt;VRF&gt;</b>	esr(config-bgp-vrf)#	CONFIG-BGP
Настройка IPv4/IPv6-адресации BGP-процесса <b>(CONFIG-BGP-FAMILY)</b>	<b>address-family { ipv4   ipv6 } unicast</b>	esr(config-bgp-af)#	CONFIG-BGP
Настройка IPv4/IPv6-адресации BGP-процесса в VRF <b>(CONFIG-BGP-VRF-FAMILY)</b>	<b>address-family { ipv4   ipv6 } unicast</b>	esr(config-bgp-vrf-af)#	CONFIG-BGP-VRF
Настройка группирования BGP-соседей <b>(CONFIG-BGP-GROUP)</b>	<b>peer-group &lt;NAME&gt;</b>	esr(config-bgp-group)#	CONFIG-BGP
Настройка группирования BGP-соседей в VRF <b>(CONFIG-BGP-VRF-GROUP)</b>	<b>peer-group &lt;NAME&gt;</b>	esr(config-bgp-vrf-group)#	CONFIG-BGP-VRF
Настройка соседа BGP-процесса <b>(CONFIG-BGP-NEIGHBOR)</b>	<b>neighbor { &lt;ADDR&gt;   &lt;IPV6-ADDR&gt; }</b>	esr(config-bgp-neighbor)#	CONFIG-BGP
Настройка соседа BGP-процесса в VRF <b>(CONFIG-BGP-VRF-NEIGHBOR)</b>	<b>neighbor { &lt;ADDR&gt;   &lt;IPV6-ADDR&gt; }</b>	esr(config-bgp-vrf-neighbor)#	CONFIG-BGP-VRF

Уровень	Команда входа <sup>1</sup>	Вид строки подсказки	Предыдущий уровень
Настройка IPv4-/IPv6-адресации BGP-соседа <b>(CONFIG-BGP-NEIGHBOR-FAMILY)</b>	<b>address-family { ipv4   ipv6   vpnv4 } unicast</b>	esr(config-bgp-neighbor-af)#	CONFIG-BGP-NEIGHBOR
Настройка IPv4-/IPv6-адресации BGP-соседа в VRF <b>(CONFIG-BGP-NEIGHBOR-FAMILY)</b>	<b>address-family { ipv4   ipv6   vpnv4 } unicast</b>	esr(config-bgp-neighbor-af-vrf)#	CONFIG-BGP-VRF-NEIGHBOR
Настройка списка подсетей <b>(CONFIG-PL)</b>	<b>ip prefix-list &lt;NAME&gt;</b>	esr(config-pl)#	CONFIG
Настройка списка подсетей <b>(CONFIG-IPV6-PL)</b>	<b>ipv6 prefix-list &lt;NAME&gt;</b>	esr(config-ipv6-pl)#	CONFIG
Настройка маршрутной карты <b>(CONFIG-ROUTE-MAP)</b>	<b>route-map &lt;NAME&gt;</b>	esr(config-route-map)#	CONFIG
Настройка правила маршрутной карты <b>(CONFIG-ROUTE-MAP-RULE)</b>	<b>rule &lt;ORDER&gt;</b>	esr(config-route-map-rule)#	CONFIG-ROUTE-MAP
Настройка RIP-протокола <b>(CONFIG-RIP)</b>	<b>router rip</b>	esr(config-rip)#	CONFIG
Настройка RIPNG-протокола <b>(CONFIG-RIPNG)</b>	<b>ipv6 router rip</b>	esr(config-ripng)#	CONFIG
Настройка IS-IS-процесса <b>(CONFIG-ISIS)</b>	<b>router isis &lt;ID&gt;</b>	esr(config-isis)#	CONFIG
Настройка OSPF-процесса <b>(CONFIG-OSPF)</b>	<b>router ospf &lt;ID&gt;</b>	esr(config-ospf)#	CONFIG
Настройка OSPF-области <b>(CONFIG-OSPF-AREA)</b>	<b>area &lt;ID&gt;</b>	esr(config-ospf-area)#	CONFIG-OSPF

Уровень	Команда входа <sup>1</sup>	Вид строки подсказки	Предыдущий уровень
Настройка виртуального соединения OSPF <b>(CONFIG-OSPF-VLINK)</b>	<b>virtual-link &lt;ID&gt;</b>	esr(config-ospf-vlink)#	CONFIG-OSPF-AREA
Настройка OSPFv3-процесса <b>(CONFIG-IPV6-OSPF)</b>	<b>ipv6 router ospf &lt;ID&gt;</b>	esr(config-ipv6-ospf)#	CONFIG
Настройка OSPFv3-области <b>(CONFIG-IPV6-OSPF-AREA)</b>	<b>area &lt;ID&gt;</b>	esr(config-ipv6-ospf-area)#	CONFIG-IPV6-OSPF
Настройка виртуального соединения OSPFv3 <b>(CONFIG-IPV6-OSPF-VLINK)</b>	<b>virtual-link &lt;ID&gt;</b>	esr(config-ipv6-ospf-vlink)#	CONFIG-IPV6-OSPF-AREA
Настройка списка ключей <b>(CONFIG-KEYCHAIN)</b>	<b>key-chain &lt;KEYCHAIN&gt;</b>	esr(config-keychain)#	CONFIG
Настройка ключа <b>(CONFIG-KEYCHAIN-KEY)</b>	<b>key &lt;ID&gt;</b>	esr(config-keychain-key)#	CONFIG-KEYCHAIN
Настройка параметров MSTP <b>(CONFIG-MST)</b>	<b>spanning-tree mst configuration</b>	esr(config-mst)#	CONFIG
Настройка правил WAN <b>(CONFIG-WAN-RULE)</b>	<b>wan load-balance rule &lt;ID&gt;</b>	esr(config-wan-rule)#	CONFIG
Настройка правил WAN (IPv6) <b>(CONFIG-IPV6-WAN-RULE)</b>	<b>ipv6 wan load-balance rule &lt;ID&gt;</b>	esr(config-ipv6-wan-rule)#	CONFIG
Настройка target-листов <b>(CONFIG-TARGET-LIST)</b>	<b>wan load-balance target-list &lt;NAME&gt;</b>	esr(config-target-list)#	CONFIG

Уровень	Команда входа <sup>1</sup>	Вид строки подсказки	Предыдущий уровень
Настройка target-листов (IPv6) <b>(CONFIG-IPV6-TARGET-LIST)</b>	<b>ipv6 wan load-balance target-list &lt;NAME&gt;</b>	esr(config-ipv6-target-list)#	CONFIG
Настройка target <b>(CONFIG-WAN-TARGET)</b>	<b>target &lt;ID&gt;</b>	esr(config-wan-target)#	CONFIG-TARGET-LIST
Настройка target (IPv6) <b>(CONFIG-IPV6-WAN-TARGET)</b>	<b>target &lt;ID&gt;</b>	esr(config-ipv6-wan-target)#	CONFIG-IPV6-TARGET-LIST
Настройка объектов отслеживания событий <b>(CONFIG-TRACK)</b>	<b>track &lt;ID&gt;</b>	esr(config-track)#	CONFIG
Настройка Wi-Fi Controller <b>(CONFIG-WIRELESS)</b>	<b>wireless-controller</b>	esr(config-wireless)#	CONFIG
Настройка VRF <b>(CONFIG-VRF)</b>	<b>ip vrf &lt;NAME&gt;</b>	esr(config-vrf)#	CONFIG
Настройка политики QoS <b>(CONFIG-POLICY-MAP)</b>	<b>policy-map &lt;NAME&gt;</b>	esr(config-policy-map)#	CONFIG
Настройка класса QoS <b>(CONFIG-CLASS-MAP)</b>	<b>class-map &lt;NAME&gt;</b>	esr(config-class-map)#	CONFIG
Настройка класса внутри политики QoS <b>(CONFIG-POLICY-MAP-CLASS)</b>	<b>class &lt;NAME&gt;</b>	esr(config-class-policy-map)#	CONFIG

Уровень	Команда входа <sup>1</sup>	Вид строки подсказки	Предыдущий уровень
Настройка PPP-пользователя для аутентификации удаленной стороны <b>(CONFIG-PPP-USER)</b>	<b>ppp chap username &lt;NAME&gt;</b>	esr(config-ppp-user)#	CONFIG-IF-E1
	<b>ppp chap username &lt;NAME&gt;</b>		CONFIG-IF-MULTILINK
	<b>user &lt;NAME&gt;</b>		CONFIG-CELLULAR-PROFILE
	<b>user &lt;NAME&gt;</b>		CONFIG-ACCESS-PROFILE
	<b>username &lt;NAME&gt;</b>		CONFIG-L2TP-SERVER
	<b>username &lt;NAME&gt;</b>		CONFIG-PPTP-SERVER
Настройка параметров резервирования конфигурации <b>(CONFIG-ARCHIVE)</b>	<b>archive</b>	esr(config-archive)#	CONFIG
Настройка сервера сбора статистики Netflow <b>(CONFIG-NETFLOW-HOST)</b>	<b>netflow collector &lt;ADDR&gt;</b>	esr(config-netflow-host)#	CONFIG
Настройка сервера сбора статистики sFlow <b>(CONFIG-SFLOW-HOST)</b>	<b>sflow collector &lt;ADDR&gt;</b>	esr(config-sflow-host)#	CONFIG
Настройка сервера получения уведомлений SNMP <b>(CONFIG-SNMP-HOST)</b>	<b>snmp-server host &lt;ADDR&gt;</b>	esr(config-snmp-host)#	CONFIG
Изменение пароля после истечения срока действия <b>(CHANGE-EXPIRED-PASSWORD)</b>	-	esr(change-expired-password)#	-
Настройка профиля фильтрации http-запросов <b>(CONFIG-PROFILE)</b>	<b>ip http profile &lt;NAME&gt;</b>	esr(config-profile)	CONFIG


Уровень	Команда входа <sup>1</sup>	Вид строки подсказки	Предыдущий уровень
Настройка общих параметров MPLS <b>(CONFIG-MPLS)</b>	<b>mpls</b>	esr(config-mpls)#	CONFIG
Настройка общих параметров LDP <b>(CONFIG-LDP)</b>	<b>ldp</b>	esr(config-ldp)#	CONFIG
Настройка IPv4-адресации протокола LDP <b>(CONFIG-LDP-AF-IPV4)</b>	<b>address-family ipv4</b>	esr(config-ldp-af-ipv4)#	CONFIG-LDP
Настройка LDP-unicast-соседства <b>(CONFIG-LDP-NEIGH)</b>	<b>neighbor &lt;ADDR&gt;</b>	esr(config-ldp-neig)#	CONFIG-LDP
Настройка IPv4-адресации протокола LDP для отдельного ip-интерфейса <b>(CONFIG-LDP-AF-IPV4-IF)</b>	<b>interface &lt;IF&gt;</b>	esr(config-ldp-af-ipv4-if)#	CONFIG-LDP-AF-IPV4
Создание L2-MPLS-туннелей <b>(CONFIG-L2VPN)</b>	<b>l2vpn</b>	esr(config-l2vpn)#	CONFIG-MPLS
Настройка L2-MPLS-туннеля <b>(CONFIG-L2VPN-EOMPLS)</b>	<b>vpls &lt;NAME&gt;</b> <b>p2p &lt;NAME&gt;</b>	esr(config-l2vpn-vpls)# esr(config-l2vpn-p2p)#	CONFIG-L2VPN
Настройка PW-class (шаблона pseudo-wire) <b>(CONFIG-L2VPN-PW-CLASS)</b>	<b>pw-class &lt;WORD&gt;</b>	esr(config-l2vpn-pw-class)#	CONFIG-L2VPN
Настройка параметров pseudo-wire L2-MPLS-туннеля <b>(CONFIG-L2VPN-PW)</b>	<b>pw &lt;PW_ID&gt; &lt;LSR_ID&gt; [&lt;NEIGH_ADDR&gt;]</b>	esr(config-l2vpn-pw)#	CONFIG-L2VPN-EOMPLS

Уровень	Команда входа <sup>1</sup>	Вид строки подсказки	Предыдущий уровень
Настройка BGP auto-discovery and signaling <b>(CONFIG-AUTODISCOVERY-BGP)</b>	<b>autodiscovery bgp</b>	esr(config-bgp)#	CONFIG-L2VPN-EOMPLS
Настройка политики IPS/IDS <b>(CONFIG-IPS-POLICY)</b>	<b>security ips policy &lt;NAME&gt;</b>	esr(config-ips-policy)#	CONFIG
Настройка поставщика правил IPS/IDS <b>(CONFIG-IPS-VENDOR)</b>	<b>vendor &lt;VENDOR&gt;</b>	esr(config-ips-vendor)#	CONFIG-IPS-POLICY
Настройка категорий правил IPS/IDS <b>(CONFIG-IPS-VENDOR-CATEGORY)</b>	<b>category &lt;CATEGORY&gt;</b>	esr(config-ips-vendor-category)#	CONFIG-IPS-VENDOR
Настройка сервиса IPS/IDS <b>(CONFIG-IPS)</b>	<b>security ips</b>	esr(config-ips)#	CONFIG
Настройка источника обновления правил IPS/IDS, распространяемых по коммерческой лицензии <b>(CONFIG-CONTENT-PROVIDER)</b>	<b>content-provider</b>	esr(config-content-provider)#	CONFIG
Настройка менеджера лицензий <b>(CONFIG-LICENCE-MANAGER)</b>	<b>licence-manager</b>	esr(config-licence-manager)	CONFIG
Настройка автообновления правил IPS/IDS из внешних источников <b>(CONFIG-IPS-AUTO-UPGRADE)</b>	<b>auto-upgrade</b>	esr(config-ips-auto-upgrade)#	CONFIG-IPS

Уровень	Команда входа <sup>1</sup>	Вид строки подсказки	Предыдущий уровень
Настройка пользовательского сервера обновлений правил IPS/IDS <b>(CONFIG-IPS-UPGRADE-USER-SERVER)</b>	<b>user-server &lt;WORD&gt;</b>	esr(config-ips-upgrade-user-server)#	CONFIG-IPS-AUTO-UPGRADE
Настройка категорий пользовательских правил IPS/IDS <b>(CONFIG-IPS-CATEGORY)</b>	<b>security ips-category user-defined &lt;CATEGORY_NAME&gt;</b>	esr(config-ips-category)#	CONFIG
Настройка пользовательских правил IPS/IDS <b>(CONFIG-IPS-CATEGORY-RULE)</b>	<b>rule &lt;ORDER&gt;</b>	esr(config-ips-category-rule)#	CONFIG-IPS-CATEGORY
Настройка расширенных пользовательских правил IPS/IDS <b>(CONFIG-IPS-CATEGORY-RULE-ADVANCED)</b>	<b>rule-advanced &lt;ORDER&gt;</b>	esr(config-ips-category-rule-advanced)#	CONFIG-IPS-CATEGORY
Настройка параметров работы zabbix-agent <b>(CONFIG-ZABBIX-AGENT)</b>	<b>zabbix-agent</b>	esr(config-zabbix-agent)#	CONFIG
Настройка параметров работы zabbix-proxy <b>(CONFIG-ZABBIX-PROXY)</b>	<b>zabbix-proxy</b>	esr(config-zabbix-proxy)#	CONFIG
Настройка почтового домена <b>CONFIG-MAILSERVER-DOMAIN</b>	<b>mailserver domain &lt;NAME&gt;</b>	esr(config-mailserver-domain)#	CONFIG
Настройка почтового сервера <b>CONFIG-MAILSERVER</b>	<b>mailserver</b>	esr(config-mailserver)#	CONFIG





Уровень	Команда входа <sup>1</sup>	Вид строки подсказки	Предыдущий уровень
Настройка локального Syslog-файла <b>CONFIG-SYSLOG-FILE</b>	<b>syslog file { flash:syslog/&lt;NAME&gt;   tmpsys:syslog/&lt;NAME&gt; }</b>	(config-syslog-file)	CONFIG
Настройка удаленного Syslog-сервера <b>CONFIG-SYSLOG-HOST</b>	<b>syslog host &lt;HOSTNAME&gt;</b>	(config-syslog-host)	CONFIG
Настройка TFTP-сервера <b>CONFIG-TFTP-SERVER</b>	<b>ip tftp server vrf &lt;VRF&gt;</b>	(config-tftp-server)	CONFIG
Настройка Firewall failover <b>CONFIG-FIREWALL-FAILOVER</b>	<b>ip firewall failover</b>	(config-firewall-failover)	CONFIG
Настройка IP failover <b>CONFIG-IP-FAILOVER</b>	<b>ip failover</b>	(config-failover)	CONFIG
Настройка Cluster <b>CONFIG-CLUSTER</b>	<b>cluster</b>	(config-cluster)	CONFIG
Настройка unit в Cluster <b>CONFIG-CLUSTER-UNIT</b>	<b>unit &lt;ID&gt;</b>	(config-cluster-unit)	CONFIG-CLUSTER




 <sup>1</sup> Подробное описание команд приведено ниже.

<sup>2</sup> Только для ESR-21.


## Типы и порядок именования интерфейсов маршрутизатора

При работе маршрутизатора используются сетевые интерфейсы различного типа и назначения. Система именования позволяет однозначно адресовать интерфейсы по их функциональному назначению и местоположению в системе. Далее в таблице приведен перечень типов интерфейсов.

Тип интерфейса	Обозначение
Физические интерфейсы	<p>Обозначение физического интерфейса включает в себя его тип и идентификатор. Идентификатор физических интерфейсов имеет вид <b>&lt;UNIT&gt;/&lt;SLOT&gt;/&lt;PORT&gt;</b>, где</p> <ul style="list-style-type: none"> <li>• <b>&lt;UNIT&gt;</b> – номер устройства в группе устройств,</li> <li>• <b>&lt;SLOT&gt;</b> – номер модуля в составе устройства или 0 при отсутствии деления устройства на модули,</li> <li>• <b>&lt;PORT&gt;</b> – порядковый номер порта.</li> </ul>
Порты 1 Гбит/с	<p><b>gigabitethernet &lt;UNIT&gt;/&lt;SLOT&gt;/&lt;PORT&gt;</b>            Пример обозначения: <b>gigabitethernet 1/0/12</b></p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p> Допускается использовать сокращенное наименование, например gi1/0/12.</p> </div>
Порты 10 Гбит/с	<p><b>tengigabitethernet &lt;UNIT&gt;/&lt;SLOT&gt;/&lt;PORT&gt;</b>            Пример обозначения: <b>tengigabitethernet 1/0/2</b></p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p> Допускается использовать сокращенное наименование, например te1/0/2.</p> </div>
Порты 25 Гбит/с	<p><b>twentyfivegigabitethernet &lt;UNIT&gt;/&lt;SLOT&gt;/&lt;PORT&gt;</b>            Пример обозначения: <b>twentyfivegigabitethernet 1/0/2</b></p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p> Допускается использовать сокращенное наименование, например twe1/0/2.</p> </div>
Порты 40 Гбит/с	<p><b>fortygigabitethernet &lt;UNIT&gt;/&lt;SLOT&gt;/&lt;PORT&gt;</b>            Пример обозначения: <b>fortygigabitethernet 1/0/2</b></p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p> Допускается использовать сокращенное наименование, например fo1/0/2.</p> </div>
Порты 100 Гбит/с	<p><b>hundredgigabitethernet &lt;UNIT&gt;/&lt;SLOT&gt;/&lt;PORT&gt;</b>            Пример обозначения: <b>hundredgigabitethernet 1/0/2</b></p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p> Допускается использовать сокращенное наименование, например hu1/0/2.</p> </div>
Порт OOB (Out-of-band)	<p><b>oob &lt;UNIT&gt;/&lt;SLOT&gt;/&lt;PORT&gt;</b>            Пример обозначения: <b>oob 1/0/1</b></p>
Группы агрегации каналов	<p>Обозначение группы агрегации каналов включает в себя его тип и порядковый номер интерфейса:</p> <p><b>port-channel &lt;CHANNEL_ID&gt;</b>            Пример обозначения: <b>port-channel 6</b></p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p> Допускается использовать сокращенное наименование, например, po1.</p> </div>

Тип интерфейса	Обозначение
Саб-интерфейсы	<p>Обозначение саб-интерфейса образуется из обозначения базового интерфейса и идентификатора (VLAN) саб-интерфейса, разделенных точкой.</p> <p>Примеры обозначений:</p> <p><b>gigabitethernet 1/0/12.100</b></p> <p><b>tengigabitethernet 1/0/2.123</b></p> <p><b>port-channel 1.6</b></p> <div style="border: 1px solid #ffc107; padding: 5px; margin-top: 10px;"> <p> Идентификатор саб-интерфейса может принимать значения [1..4094].</p> </div>
Q-in-Q интерфейсы	<p>Обозначение Q-in-Q интерфейса образуется из обозначения базового интерфейса, идентификатора сервисного VLAN и идентификатора пользовательского VLAN, разделенных точкой.</p> <p>Примеры обозначений:</p> <p><b>gigabitethernet 1/0/12.100.10</b></p> <p><b>tengigabitethernet 1/0/2.45.12</b></p> <p><b>port-channel 1.6.34</b></p> <div style="border: 1px solid #ffc107; padding: 5px; margin-top: 10px;"> <p> Идентификатор сервисного и пользовательского VLAN может принимать значения [1..4094].</p> </div>
E1-интерфейсы	<p>Обозначение E1-интерфейса включает в себя его тип и идентификатор.</p> <p>Идентификатор E1-интерфейсов имеет вид <b>&lt; UNIT&gt;/&lt; SLOT&gt;/&lt; STREAM&gt;</b>, где</p> <ul style="list-style-type: none"> <li>• <b>&lt;UNIT&gt;</b> – номер устройства в группе устройств [1..2],</li> <li>• <b>&lt;SLOT&gt;</b> – номер E1-модуля в составе устройства [0..12],</li> <li>• <b>&lt;STREAM&gt;</b> – порядковый номер E1-потока [1..1].</li> </ul> <p>Пример обозначения: <b>e1 1/0/1</b></p> <div style="border: 1px solid #dc3545; padding: 5px; margin-top: 10px;"> <p> На маршрутизаторах ESR-1000 не поддерживается работа модулей ToPGATE-WAN-E1 с аппаратной версией (hardware revision) 812.</p> </div>
Группы агрегации E1-каналов	<p>Обозначение группы агрегации E1-каналов включает в себя его тип и порядковый номер интерфейса:</p> <p><b>multilink &lt;CHANNEL_ID&gt;</b></p> <p>Пример обозначения: <b>multilink 3</b></p>
Логические интерфейсы	<p>Обозначение логического интерфейса является порядковым номером интерфейса:</p> <p>Примеры обозначений:</p> <p><b>loopback 4</b></p> <p><b>bridge 60</b></p> <p><b>service-port 1</b></p>

Тип интерфейса	Обозначение
Последовательные интерфейсы	<p>Обозначение последовательного интерфейса включает в себя его тип и идентификатор. Идентификатор serial интерфейсов и line-aux интерфейсов имеет вид <b>&lt; UNIT&gt;/&lt; SLOT&gt;/&lt; PORT&gt;</b>, где</p> <ul style="list-style-type: none"> <li>• <b>&lt;UNIT&gt;</b> – номер устройства в группе устройств [1..2],</li> <li>• <b>&lt;SLOT&gt;</b> – номер модуля в составе устройства или 0 при отсутствии деления устройства на модули,</li> <li>• <b>&lt;PORT&gt;</b> – порядковый номер порта.</li> </ul> <p>Пример обозначения: <b>serial 1/0/1, line aux 1/0/1</b></p>
USB-модемы	<p>Обозначение USB-модема включает в себя его тип и порядковый номер:</p> <p><b>modem &lt;MODEM-NUM&gt;</b></p> <p>Пример обозначения: <b>modem 1</b></p>
FXS/FXO-порты	<p>Обозначение FXS/FXO-портов включает в себя его тип и порядковый номер:</p> <p><b>interface voice-port &lt;NUM&gt;</b></p> <p>Пример обозначения: <b>voice-port 1</b></p>

-  1. Количество интерфейсов каждого типа зависит от модели маршрутизатора.  
 2. Текущая версия ПО поддерживает кластеризацию устройств единой модели. Номер unit в группе устройств может принимать значение 1 или 2.  
 3. Некоторые команды поддерживают одновременную работу с группой интерфейсов. Для указания группы интерфейсов может быть использовано перечисление через запятую или указание диапазона идентификаторов через дефис «-».

Примеры указания групп интерфейсов:

```
interface gigabitethernet 1/0/1, gigabitethernet 1/0/5
interface tengigabitethernet 1/0/1-2
interface fortygigabitethernet 1/0/1-2
interface gil/0/1-3, gil/0/7, te1/0/1, fo1/0/1
```

Таблица 4 – Типы и порядок именования интерфейсов маршрутизатора

### Типы и порядок именования туннелей маршрутизатора


При работе маршрутизатора используются сетевые туннели различного типа и назначения. Система именования позволяет однозначно адресовать туннели по их функциональному назначению. Далее в таблице приведен перечень типов туннелей.

Таблица 5 – Типы и порядок именования туннелей маршрутизатора

Тип туннеля	Обозначение
L2TP-туннель	<p>Обозначение L2TP-туннеля состоит из обозначения типа и порядкового номера туннеля:</p> <p><b>l2tp &lt;L2TP_ID&gt;</b></p> <p>Пример обозначения: <b>l2tp 1</b></p>

Тип туннеля	Обозначение
L2TPv3-туннель	Обозначение L2TPv3-туннеля состоит из обозначения типа и порядкового номера туннеля: <b>l2tpv3 &lt;L2TPV3_ID&gt;</b> Пример обозначения: <b>l2tpv3 1</b>
GRE-туннель	Обозначение GRE-туннеля состоит из обозначения типа и порядкового номера туннеля: <b>gre &lt;GRE_ID&gt;</b> Пример обозначения: <b>gre 1</b>
GRE саб-туннель	Обозначение GRE саб-туннеля состоит из обозначения типа, порядкового номера туннеля и идентификатора VLAN для саб-туннеля: <b>gre &lt;GRE_ID&gt;.&lt;VLAN_ID&gt;</b> Пример обозначения: <b>gre 1.200</b>
SoftGRE-туннель	Обозначение SoftGRE-туннеля состоит из обозначения типа, порядкового номера туннеля и, опционально, VLAN ID виртуального интерфейса: <b>softgre &lt;GRE_ID&gt;[.&lt;VLAN&gt;]</b> Примеры обозначения: <b>softgre 1, softgre 1.10</b>
IPv4-over-IPv4-туннель	Обозначение IPv4-over-IPv4-туннеля состоит из обозначения типа и порядкового номера туннеля: <b>ip4ip4 &lt;IPIP_ID&gt;</b> Пример обозначения: <b>ip4ip4 1</b>
IPsec-туннель	Обозначение виртуального IPsec-туннеля состоит из обозначения типа и порядкового номера туннеля: <b>vti &lt;VTI_ID&gt;</b> Пример обозначения: <b>vti 1</b>
Логический туннель (туннель между VRF)	Обозначение логического туннеля состоит из обозначения типа и порядкового номера туннеля: <b>lt &lt;LT_ID&gt;</b> Пример обозначения: <b>lt 1</b>
PPPoE-туннель	Обозначение PPPoE-туннеля состоит из обозначения типа и порядкового номера туннеля: <b>pppoe &lt;PPPoE_ID&gt;</b> Пример обозначения: <b>pppoe 1</b>
PPTP-туннель	Обозначение PPTP-туннеля состоит из обозначения типа и порядкового номера туннеля: <b>pptp &lt;PPTP_ID&gt;</b> Пример обозначения: <b>pptp 1</b>

Тип туннеля	Обозначение
OpenVPN-туннель	Обозначение OpenVPN-туннеля состоит из обозначения типа и порядкового номера туннеля: <b>openvpn &lt;OPENVPN_ID&gt;</b> Пример обозначения: <b>openvpn 1</b>
WireGuard-туннель	Обозначение WireGuard-туннеля состоит из обозначения типа и порядкового номера туннеля: <b>wireguard &lt;WG_ID&gt;</b> Пример обозначения: <b>wireguard 1</b>

 Количество туннелей каждого типа зависит от модели и ПО маршрутизатора.

## 5 Команды пользовательского интерфейса

- alarm
- clear alarms
- clear ssh host
- configure
- do
- end
- exit
- help
- history size
- logout
- monitor
- ping
- reload cancel
- reload system
- show alarms brief
- show alarms brief active
- show alarm settings
- show history
- show system reload
- show tech-support
- ssh
- telnet
- terminal datadump
- terminal mode
- terminal resize
- traceroute
- uptime
- verify
- verify filesystem
- verify storage-device

### alarm

Данной командой устанавливаются значения порогов аварий.

#### Синтаксис

```
alarm { facility <FACILITY-ALARMS> | memory <MEMORY-ALARMS> | process <PROCESS-ALARMS> } <VALUE>
```

```
no alarm { facility <FACILITY-ALARMS> | memory <MEMORY-ALARMS> | process <PROCESS-ALARMS> } <VALUE>
```

#### Параметры

<FACILITY-ALARMS> – пороги аварий, связанных с окружением (скорость FAN, температура процессора и датчиков). Принимает следующие значения:

- fan-speed high – порог скорости вращения вентилятора, выраженный в процентах от максимальной скорости вращения (0..100), при переходе через который от меньших значений к большему, генерируется SNMP trap eltexEnvFanSpeedHigh (авария). Значение этого порога должно быть больше, чем alarm facility fan-speed low (для ESR-100/200/1000/1200/1500/1700/3100/3200/3200L/3300);

- fan-speed low – порог скорости вращения вентилятора, выраженный в процентах от максимальной скорости вращения (0..100), при переходе через который от больших значений к меньшим, генерируется SNMP trap eltexEnvFanSpeedHighOk (нормализация аварии). Значение этого порога должно быть меньше, чем alarm facility fan-speed high (для ESR-100/200/1000/1200/1500/1511/1700/3100/3200/3200L/3300);
- temperature cpu critical high – порог температуры процессора, выраженный в градусах по шкале Цельсия (0..100), при переходе через который от меньших значений к большим, генерируется SNMP trap eltexEnvTempCritical (авария). Значение этого порога должно быть самым большим из всех значений температурных порогов процессора;
- temperature cpu critical low – порог температуры процессора, выраженный в градусах по шкале Цельсия (0..100), при переходе через который от больших значений к меньшим, генерируется SNMP trap eltexEnvTempCriticalOk (нормализация аварии). Значение этого порога должно быть меньше, чем alarm facility temperature cpu critical high, но больше, чем alarm facility temperature cpu overheat high;
- temperature <sensor> overheat high – порог температуры сенсора, выраженный в градусах по шкале Цельсия (0..100), при переходе через который от меньших значений к большим, генерируется SNMP trap eltexEnvTempOverheat (авария). Значение этого порога должно быть меньше, чем alarm facility temperature <sensor> critical low, но больше, чем alarm facility temperature <sensor> overheat low;
- temperature <sensor> overheat low – порог температуры сенсора, выраженный в градусах по шкале Цельсия (0..100), при переходе через который от больших значений к меньшим, генерируется SNMP trap eltexEnvTempOverheatOk (нормализация аварии). Значение этого порога должно быть меньше, чем alarm facility temperature <sensor> overheat high;
- temperature <sensor> supercooling high – порог температуры сенсора, выраженный в градусах по шкале Цельсия (0..12), при переходе через противоположное значение которого от меньших значений к большим, генерируется SNMP trap eltexEnvTempSupercoolingOk (нормализация аварии). Этот порог предназначен для задания отрицательных температур, и его значение должно быть меньше, чем alarm facility temperature <sensor> supercooling low;
- temperature <sensor> supercooling low – порог температуры сенсора, выраженный в градусах по шкале Цельсия (0..15), при переходе через противоположное значение которого от больших значений к меньшим, генерируется SNMP trap eltexEnvTempSupercooling (авария). Этот порог предназначен для задания отрицательных температур, и его значение должно быть больше, чем alarm facility temperature <sensor> supercooling high;

Поле сенсор может принимать следующие значения для каждой модели маршрутизатора:

- ESR-10: cpu;
- ESR-12V: cpu, switch;
- ESR-12VF/15R/15VF: cpu, switch, sfp;
- ESR-20/21/30/31: cpu;
- ESR-100/200: cpu, board;
- ESR-1000/1200/1500/1511/1700: cpu, board, switch, sfp;
- ESR-3100/3200/3200L/3300: cpu, board, sfp, phy.

<MEMORY-ALARMS> – пороги аварий, связанных со свободным пространством NAND и RAM. Принимает следующие значения:

- free low-watermark flash high – порог количества свободной flash-памяти, выраженной в процентах от максимального (0..100), при переходе через который от меньших значений к большим генерируется SNMP trap eltexEnvMemoryLowOk (нормализация аварии). Значение этого порога должно быть самым большим из всех значений порогов количества свободной flash-памяти;
- free low-watermark flash low – порог количества свободной flash-памяти, выраженной в процентах от максимального (0..100), при переходе через который от больших значений к меньшим генерируется SNMP trap eltexEnvMemoryLow (авария). Значение этого порога должно быть меньше, чем alarm memory free low-watermark flash high;
- free low-watermark ram high – порог количества свободной RAM-памяти, выраженной в процентах от максимального (0..100), при переходе через который от меньших значений к большим генерируется SNMP trap eltexEnvMemoryLowOk (нормализация аварии). Значение этого порога должно быть самым большим из всех значений порогов количества свободной RAM-памяти;



- free low-watermark ram low – порог количества свободной RAM-памяти, выраженной в процентах от максимального (0..100), при переходе через который от больших значений количества к меньшим генерируется SNMP trap eltexEnvMemoryLow (авария). Значение этого порога должно быть меньше, чем alarm memory free low-watermark ram high;
- reserve critical flash high – порог количества свободной flash-памяти, выраженной в процентах от максимального (0..100), при переходе через который от меньших значений к большим генерируется SNMP trap eltexEnvMemoryCriticalLowOk (нормализация аварии). Значение этого порога должно быть меньше, чем alarm memory free low-watermark flash low, но больше, чем alarm memory reserve critical flash low;
- reserve critical flash low – порог количества свободной flash-памяти, выраженной в процентах от максимального (0..100), при переходе через который от больших значений количества к меньшим генерируется SNMP trap eltexEnvMemoryCriticalLow (авария). Значение этого порога должно быть меньше, чем alarm memory reserve critical flash high;
- reserve critical ram high – порог количества свободной RAM-памяти, выраженной в процентах от максимального (0..100), при переходе через который от меньших значений к большим генерируется SNMP trap eltexEnvMemoryCriticalLowOk (нормализация аварии). Значение этого порога должно быть меньше, чем alarm memory free low-watermark ram low, но больше, чем alarm memory reserve critical ram low;
- reserve critical ram low – порог количества свободной RAM-памяти, выраженной в процентах от максимального (0..100), при переходе через который от больших значений количества к меньшим генерируется SNMP trap eltexEnvMemoryCriticalLow (авария). Значение этого порога должно быть меньше, чем alarm memory reserve critical ram high.

<PROCESS-ALARMS> – пороги аварий, связанных с загрузкой процессора. Принимает следующие значения:

- cpu threshold falling – порог утилизации CPU, выраженной в процентах от максимальной (0..100), при переходе через который от больших значений к меньшим генерируется SNMP trap eltexEnvCpuLoadHighOk (нормализация аварии). Значение этого порога должно быть меньше, чем alarm process cpu threshold rising;
- cpu threshold rising – порог утилизации CPU, выраженной в процентах от максимальной (0..100), при переходе через который от меньших значений к большим генерируется SNMP trap eltexEnvCpuLoadHigh (авария). Значение этого порога должно быть больше, чем alarm process cpu threshold falling.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример:

```
esr(config)# alarm facility fan-speed high 80
```

## clear alarms

Данной командой осуществляется очистка записей об авариях.

## Синтаксис

```
clear alarms { inactive | all }
```

## Параметры

inactive – удаляются записи только об архивных авариях;

all – удаляются записи об архивных и активных авариях.

## Необходимый уровень привилегий

10

## Командный режим

ROOT

## Пример:

```
esr# clear alarms inactive
```

## clear ssh host

Данной командой осуществляется сброс сохраненного SSH-ключа удаленного хоста.

## Синтаксис

```
clear ssh host { <ADDR> | <IPV6-ADDR> | <HOST> } [ port <PORT>]
```

## Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<HOST> – доменное имя, задается строкой до 254 символа;

<PORT> – номер TCP-порта, принимает значения [1..65535];

## Необходимый уровень привилегий

15

## Командный режим

ROOT

## Пример:

```
esr# clear ssh host 192.168.1.1
```

## configure

Данная команда позволяет перейти в режим глобального конфигурирования.

## Синтаксис

```
configure  
configure terminal
```

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

10

## Командный режим

ROOT

## Пример:

```
esr# configure  
esr(config)#
```

## do

Команда do позволяет выполнять команды корневого режима (ROOT) из любого другого режима командного интерфейса.

## Синтаксис

```
do <command>
```

## Параметры

<command> – команда корневого режима.

## Необходимый уровень привилегий

1

## Командный режим

Все режимы, кроме корневого.

## Пример:

```
esr(config)# do show version  
Boot version:  
  1.24.0.1 (2024-08-01 13:40:59)  
SW version:  
  1.24.0 build 1[d9bdbda] (2024-08-01 18:06:41)  
HW version:  
  1v7
```

**end**

Команда служит для возврата в корневой командный режим (ROOT).

**Синтаксис**

end

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

1

**Командный режим**

Все режимы, кроме корневого.

**exit**

Данная команда служит для возврата на уровень выше в иерархической системе командных режимов.

При выполнении данной команды в режиме ROOT завершается сеанс работы пользователя с интерфейсом командной строки CLI маршрутизатора.

**Синтаксис**

exit

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

1

**Командный режим**

Все режимы.

**help**

Данной командой на дисплей выводится информация о работе с командной строкой.

**Синтаксис**

help

**Параметры**

Команда не содержит параметров.

## Необходимый уровень привилегий

1

## Командный режим

Все режимы.

## history size

Данной командой можно изменить максимальное число последних введенных команд пользователя, которые сохраняются в истории команд текущей сессии. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
history size <SIZE>  
no history size
```

## Параметры

<SIZE> – число последних введенных команд, принимает значения [10..1000].

## Значение по умолчанию

50

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример:

```
esr# history size 20
```

## logout

Данной командой завершается сеанс работы пользователя с интерфейсом командной строки CLI.

## Синтаксис

```
logout
```

## Параметры

Команда не содержит параметров

## Необходимый уровень привилегий

1

## Командный режим

ROOT

CHANGE-EXPIRED-PASSWORD

## Пример:

```
esr# logout
```

## monitor

Данной командой включается мониторинг трафика на сетевом интерфейсе в режиме реального времени по пакетно.

Реализована запись дампа трафика в файл (.pcap, .pcapng, .txt, ...) с последующей возможностью копирования на usb/mms/flash:data/tftp-server. Записанный в файл дамп трафика в раздел файловой системы flash:data/ лимитирован 1000 пакетами.

## Синтаксис

```
monitor { <IF> | <TUN> } [ protocol <TYPE> [ source-port <SRC-PORT> ] [ destination-port <DST-PORT> ] [ port <PORT> ] ] [ source-address {<SRC-ADDR> | <SRC-IPV6-ADDR> } ] [ destination-address { <DST-ADDR> | <DST-IPV6-ADDR> } ] [address { <ADDR> | <IPV6-ADDR> } ] [ packets <VALUE> ] [ detailed ] [ file <META-PATH> ] [ detailed ]
```

## Параметры

<IF> – интерфейс или группа интерфейсов, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<TUN> – имя туннеля, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#);

<SRC-ADDR> – IP-адрес отправителя, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<DST-ADDR> – IP-адрес получателя, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<SRC-IPV6-ADDR> – IPv6-адрес отправителя, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<DST-IPV6-ADDR> – IPv6-адрес получателя, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<ADDR> – IP-адрес отправителя или получателя, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес отправителя или получателя, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<SRC-MAC> – MAC-адрес отправителя ethernet-кадра, задается в виде HH:HH:HH:HH:HH:HH, где каждая HH-часть принимает значения в шестнадцатеричном формате [0..FF];

<DST-MAC> – MAC-адрес получателя ethernet-кадра, задается в виде HH:HH:HH:HH:HH:HH, где каждая HH-часть принимает значения в шестнадцатеричном формате [0..FF];

<SRC-MAC> – MAC-адрес отправителя или получателя ethernet-кадра, задается в виде HH:HH:HH:HH:HH:HH, где каждая HH-часть принимает значения в шестнадцатеричном формате [0..FF];

<TYPE> – тип протокола, принимает значения: tcp, udp, icmp, icmp6, igmp, igmp, arp, gre, ipip, esp, ah, eigrp, ospf, pim, vrrp, l2tp, RDP или номер протокола [0..255];

<SRC-PORT> – номер TCP/UDP-порта отправителя, принимает значения [1..65535];

<DST-PORT> – номер TCP/UDP-порта получателя, принимает значения [1..65535];

<PORT> – номер TCP/UDP-порта отправителя или получателя, принимает значения [1..65535];

<VALUE> – количество пакетов, после получения которых анализ будет остановлен, указывается в диапазоне [1...4294967295];

detailed – информация выдается в детализированном формате.

<FILE> – запись дампа трафика в файл с любым расширением – .pcap, .pcapng, .txt и др.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример:

```
esr# monitor gigabitethernet 1/0/5 detailed
23:37:44.324049 d8:50:e6:d2:f0:46 > a8:f9:4b:aa:03:a5, ethertype IPv4 (0x0800), length 98: (tos
0x0, ttl
64, id 50760, offset 0, flags [DF], proto ICMP (1), length 84)
10.255.100.1 > 10.255.100.5: ICMP echo request, id 11730, seq 19, length 64
esr# monitor gigabitethernet 1/0/5 protocol tcp file flash:data://dump.pcap
esr# copy flash:data://dump.pcap tftp://<ip-address>:<port>:/copied_dump.pcap
```

### ping

Данная команда используется для проверки доступности указанного сетевого устройства.

### Синтаксис

```
ping [ vrf <VRF> ] { <ADDR> | <IPV6-ADDR> | <HOSTNAME> [ { ip | ipv6 } ] [ ttl <TTL> ]
[ packets <COUNT> | unlimited ] [ size <SIZE> ] [ timeout <TIMEOUT> ] [ interval
<INTERVAL> ] [ source { ip { <SRC-ADDR> | <SRC-IPV6-ADDR> } | interface <IF> | tunnel
<TUN> } ] [ data <HEX> ] [ dscp <DSCP> ] [ flood ] [ detailed ] [ strategy <STRATEGY> ]
[ nodeinfo <INFO> ] [ broadcast ]
```

### Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа;

<ADDR> – IP-адрес устройства, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPv6-ADDR> – IPv6-адрес устройства, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<HOSTNAME> – DNS-имя устройства, задаётся строкой до 255 символов;

- ip – проверять доступность при помощи ipv4-пакетов;
- ipv6 – проверять доступность при помощи ipv6-пакетов.

<TTL> – время жизни IP-пакета, принимает значение [1..255], по умолчанию 64;

<COUNT> – количество передаваемых пакетов [1..4294967295], по умолчанию 5;

<SIZE> – размер icmp-пакета в байтах, принимает значение [1..65468], по умолчанию 56 байт, что соответствует 64 байтам после добавления заголовка ICMP и 84 байтам после добавления ip-заголовка;

<TIMEOUT> – время ожидания ответа, в секундах. Опция влияет на таймаут, если отсутствуют какие-либо ответы, в противном случае утилита ждёт два RTTs. Принимает значение [1..60], по умолчанию 1 секунда;

<INTERVAL> – интервал между отправками icmp-пакетов в миллисекундах, принимает значение [200..60000], по умолчанию 1000.

<SRC-ADDR> – IP-адрес отправителя, в качестве данного адреса может использоваться любой IP-адрес маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<SRC-IPv6-ADDR> – IPv6-адрес отправителя, в качестве данного адреса может использоваться любой IPv6-адрес маршрутизатора, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<IF> – имя интерфейса маршрутизатора, через который будут отправлены пакеты, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<TUN> – имя туннеля маршрутизатора, через который будут отправлены пакеты, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#);

<HEX> – шаблон данных, которым будет заполняться пакет, задаётся числом в шестнадцатеричной системе до 16 байт;

<DSCP> – DSCP-приоритет в соответствии с RFC 2474, принимает значение [0..63], значение по умолчанию 0;

flood – при указании данной команды пакеты будут отправляться с максимальной скоростью, ответы от устройства не отображаются до окончания выполнения команды;

broadcast – при указании данной команды будет разрешено отправлять пакеты на широковещательный адрес;

detailed – при указании данного ключа будет выводиться полная информация о icmp-ответах и общая статистика работы команды. Без указания – только статистика.

<STRATEGY> – стратегия фрагментации пакетов, принимает одно из следующих значений:

- allow-fragmentation – разрешить фрагментацию, не устанавливать флаг DF (don't fragment);
- discovery-pmtu – выполнять изучение PMTU (Path MTU), фрагментировать локально, если размер пакета слишком большой;
- disallow-fragmentation – запретить фрагментацию, в том числе локальную.

<INFO> – только для IPv6. Отправка ICMPv6 Node Information Queries (RFC4620), вместо Echo Request принимает одно из следующих значений:



- name – запрос DNS-имен(и) узла;
- ipv6 – запрос IPv6-адресов узла;
- ipv6-global – запрос глобальных IPv6-адресов узла;
- ipv6-sitelocal – запрос site-local IPv6-адресов узла;
- ipv6-linklocal – запрос link-local IPv6-адресов узла;
- ipv6-all – запрос unicast IPv6-адресов узла;
- ipv4 – запрос IP-адресов узла;
- ipv4-all – запрос IP-адресов со всех сетевых интерфейсов узла.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример:

```

esr# ping 192.168.100.39 packets 5 size 1400 detailed
PING 192.168.100.39 (192.168.100.39) 1400(1428) bytes of data.
1408 bytes from 192.168.100.39: icmp_req=1 ttl=64 time=0.084 ms
1408 bytes from 192.168.100.39: icmp_req=2 ttl=64 time=0.053 ms
1408 bytes from 192.168.100.39: icmp_req=3 ttl=64 time=0.082 ms
1408 bytes from 192.168.100.39: icmp_req=4 ttl=64 time=0.051 ms
1408 bytes from 192.168.100.39: icmp_req=5 ttl=64 time=0.075 ms
--- 192.168.100.39 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 3999ms
rtt min/avg/max/mdev = 0.051/0.069/0.084/0.014 ms
esr# ping ipv6 fc00::1
PING fc00::1(fc00::1) 56 data bytes
64 bytes from fc00::1: icmp_seq=1 ttl=64 time=0.379 ms
64 bytes from fc00::1: icmp_seq=2 ttl=64 time=0.161 ms
--- fc00::1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1000ms
rtt min/avg/max/mdev = 0.161/0.270/0.379/0.109 ms

```

### reload cancel

Данной командой осуществляется отмена ранее запланированной перезагрузки устройства.

### Синтаксис

```
reload cancel
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

15

**Командный режим**

ROOT

**Пример:**

```
esr# reload cancel
```

**reload system**

Данной командой осуществляется перезагрузка устройства.

**Синтаксис**

```
reload system { in { <MINUTES> | <HOUR>:<MINUTES>} | at <TIME> <DATE> }
```

**Параметры**

<HOUR> – количество часов до перезагрузки,

<MINUTES> – количество минут до перезагрузки,

<TIME> – точное время перезагрузки в формате HH:MM:SS:

HH – часы в диапазоне [0..23],

MM – минуты в диапазоне [0..59],

SS – секунды в диапазоне [0..59].

<DATE> – дата перезагрузки в формате DAY MONTH YEAR:

DAY – день перезагрузки, принимает значение в диапазоне [1..31],

MONTH – месяц перезагрузки, принимает одно из значений:

- january
- february
- march
- april
- may
- june
- july
- august
- september
- october
- november
- december

YEAR – год перезагрузки, принимает значение в диапазоне [2001..2037].

**Необходимый уровень привилегий**

15

**Командный режим**

ROOT

**Пример:**

```
esr# reload system
```

**show alarms brief**

Данной командой выводится краткая история аварий на маршрутизаторе (активных и завершенных).

**Синтаксис**

```
show alarms brief
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

10

**Командный режим**

ROOT

**Пример:**

```
esr# show alarms brief

  History Alarms
  ~~~~~
Severity   Group           Set time           Clear time          Description
-----
major     environment     2000-03-31 17:27:38 2000-03-31 17:31:53 Reserve Power Supply
Fault
```

**show alarms brief active**

Данной командой выводится краткая информация о текущих (активных) авариях на маршрутизаторе.

**Синтаксис**

```
show alarms brief active
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример:**

```

esr# show alarms brief active
  History Alarms
  ~~~~~
Severity   Group           Set time           Clear time          Description
-----
notify     environment     2000-03-31 16:47:05 -                    Reserve Power Supply
removed

```

**show alarm settings**

Данной командой выводится информация о настройках порогов срабатывания аварий.

**Синтаксис**

```
show alarm settings <TYPE>
```

**Параметры**

<TYPE> – тип аварий:

- facility – пороги аварий, связанных с окружением (скорость FAN, температура процессора и датчиков);
- memory – пороги аварий, связанных с свободным пространством NAND и RAM;
- process – пороги аварий, связанных с загрузкой процессора.

**Необходимый уровень привилегий**

10

**Командный режим**

ROOT

**Пример:**

```

esr# show alarm facility
fan-speed:
  high:          80
  low:           75
temperature CPU:
  critical high: 95
  critical low:  92
  overheat high: 80
  overheat low:  78
  supercooling high: -12
  supercooling low: -15
temperature sensor1:

```

```

overheat high:          60
overheat low:           57
supercooling high:     -12
supercooling low:      -15
temperature sensor2:
overheat high:          60
overheat low:           57
supercooling high:     -12
supercooling low:      -15
temperature sensor3:
overheat high:          60
overheat low:           57
supercooling high:     -12
supercooling low:      -15

```

## show history

Данной командой на дисплей выводится информация о командах, которые использовались в текущей сессии, или о количестве сохраняемых команд.

### Синтаксис

```
show history [size | <NUM> ]
```

### Параметры

size – максимальное число последних введенных команд пользователя, которые сохранятся в истории команд текущей сессии.

<NUM> – количество последних введенных команд пользователя, которые необходимо отобразить [0..100].

### Значение по умолчанию

0. Соответствует выводу всей истории введенных команд.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример:

```

esr# show history
 1 enable
 2 show history
 3 configure
 4 service nat
 5 service nat source
 6 exit
 7 show history

```

## show system reload

Данной командой на дисплей выводится информация о командах, дате и времени запланированной перезагрузки.

### Синтаксис

```
show system reload
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример:

```
esr# show system reload
System reload is scheduled for 2022-08-10 13:02:02
```

## show tech-support

Данная команда используется для сбора комплексной системной информации с устройства.

### Синтаксис

```
show tech-support
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример:

```
esr# show tech-support
|*****| 100% Success.
Show tech-support output available at: flash:data/20110413_112242_show_tech-supz
```

The created archive may contain **private** data. You can delete them yourself  
Execute time 58.995103 sec

## ssh

Данная команда используется для подключения к удаленному узлу по протоколу SSH.

### Синтаксис

```
ssh [ vrf <VRF> ] <USERNAME> { <ADDR> | <IPV6-ADDR> | <HOSTNAME> } [ port <PORT> ]
[ version <VERSION> ] [ source { <SRC-ADDR> | <SRC-IPV6-ADDR> } ] [ dscp <DSCP> ]
```

### Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа;

<USERNAME> – имя пользователя, задаётся строкой до 31 символа;

<ADDR> – IP-адрес устройства, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес устройства, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<HOSTNAME> – DNS-имя устройства, задаётся строкой до 253 символов;

<PORT> – номер TCP-порта, прослушиваемого SSH-сервером, принимает значения [1..65535]. По умолчанию установлено 22;

<VERSION> – версия SSH-протокола, принимает значения [1..2]. По умолчанию используется версия 1;

<SRC-ADDR> – IP-адрес отправителя, в качестве данного адреса может использоваться любой IP-адрес маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<SRC-IPV6-ADDR> – IPv6-адрес отправителя, в качестве данного адреса может использоваться любой IPv6-адрес маршрутизатора, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<DSCP> – DSCP-приоритет в соответствии с RFC 2474, принимает значение [0..63], значение по умолчанию 0.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример:

```
esr# ssh tester 10.100.100.1
The authenticity of host '10.100.100.1 (10.100.100.1)' can't be established.
ECDSA key fingerprint is db:e4:0a:93:59:87:7d:9f:90:5c:19:a3:e7:97:ec:d5.
Are you sure you want to continue connecting (yes/no)? yes
%AAA-I-SSH: Warning: Permanently added '10.100.100.1' (ECDSA) to the list of known hosts.
tester@10.100.100.1's password:
Welcome to Ubuntu 14.04.2 LTS (GNU/Linux 3.13.0-51-generic x86_64)
 * Documentation:  https://help.ubuntu.com/
System information as of Mon May 25 09:25:10 NOVT 2015
```

```
Last login: Tue May 12 19:39:11 2015
(tester@kubuntu ~) $
```

## telnet

Данная команда используется для подключения к удаленному узлу по протоколу Telnet.

### Синтаксис

```
telnet [ vrf <VRF> ] { <ADDR> | <IPV6-ADDR> | <HOSTNAME> } [ port <PORT> ] [ source
{ <SRC-ADDR> | <SRC-IPV6-ADDR> } ] [ dscp <DSCP> ]
```

### Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа;

<ADDR> – IP-адрес устройства, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес устройства, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<HOSTNAME> – DNS-имя устройства, задаётся строкой до 253 символов;

<PORT> – номер TCP-порта, прослушиваемого SSH-сервером, принимает значения [1..65535], по умолчанию 23;

<SRC-ADDR> – IP-адрес отправителя, в качестве данного адреса может использоваться любой IP-адрес маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<SRC-IPV6-ADDR> – IPv6-адрес отправителя, в качестве данного адреса может использоваться любой IPv6-адрес маршрутизатора, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<DSCP> – DSCP-приоритет в соответствии с RFC 2474, принимает значение [0..63], значение по умолчанию 0.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример:

```
esr# telnet 10.100.100.1
Entering character mode
Escape character is '^]'.
Ubuntu 14.04.2 LTS
kubuntu login: tester
Password:
Last login: Mon May 25 15:23:06 NOVT 2015 from sw31-1.eltex.loc on pts/16
Welcome to Ubuntu 14.04.2 LTS (GNU/Linux 3.13.0-51-generic x86_64)
 * Documentation:  https://help.ubuntu.com/
   System information as of Mon May 25 15:23:01 NOVT 2015
(tester@kubuntu ~) $
```



## terminal datadump

Команда используется для выключения постраничного режима вывода трассировок для текущей сессии.

Использование отрицательной команды включает постраничный режим вывода трассировок.

### Синтаксис

```
[no] terminal datadump
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример:

```
esr# terminal datadump
```

## terminal mode

Команда используется для переключения режима работы терминала для текущей сессии.

Использование отрицательной команды включает человекочитаемый режим работы терминала.

### Синтаксис

```
terminal mode { human | machine }
```

### Параметры

human – человекочитаемый режим работы терминала, содержимое терминала перерисовывается при каждом пользовательском вводе, предназначен для интерактивного использования;

machine – машинный режим работы терминала, содержимое терминала не перерисовывается при пользовательском вводе, предназначен для взаимодействия с CLI через скрипты.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

**Пример:**

```
esr# terminal mode machine
```

**terminal resize**

Команда используется для масштабирования размера терминала под размер окна при использовании консольного подключения.

**Синтаксис**

```
terminal resize
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример:**

```
esr# terminal resize
```

**traceroute**

Данная команда используется для трассировки маршрута до указанного сетевого устройства.

**Синтаксис**

```
traceroute [ vrf <VRF> ] { <ADDR> | <IPV6-ADDR> | <HOSTNAME> [ { ip | ipv6 } ] }
[ first-ttl <FIRST-TTL> ] [ max-ttl <MAX-TTL> ] [ timeout <TIMEOUT> ] [ source { ip
{ <SRC-ADDR> | <SRC-IPV6-ADDR> } | interface <IF> | tunnel <TUN> } ] [ dscp <DSCP> ]
[ protocol { icmp | udp [ <PORT> ] | tcp [ <PORT> ] } ] [ gateway { <GW-ADDR> | <GW-IPV6-
ADDR> } ]
```

**Параметры**

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа;

<ADDR> – IP-адрес устройства, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес устройства, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<HOSTNAME> – DNS-имя устройства, задаётся строкой до 255 символов;

- ip – проводить трассировку при помощи ipv4-пакетов;

- ipv6 – проводить трассировку при помощи ipv6-пакетов.

<FIRST-TTL> – время жизни IP-пакета, значение, с которого начинается трассировка маршрута, принимает значение [1..255], по умолчанию 1;

<MAX-TTL> – время жизни IP-пакета, значение, на котором заканчивается трассировка маршрута, принимает значение [1..255], по умолчанию 30;

<TIMEOUT> – время ожидания ответа, в секундах. Опция влияет на таймаут, если отсутствуют какие-либо ответы, в противном случае утилита ждет два RTTs. Принимает значение [1..60], по умолчанию 5 секунд;

<SRC-ADDR> – IP-адрес отправителя, в качестве данного адреса может использоваться любой IP-адрес маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<SRC-IPV6-ADDR> – IPv6-адрес отправителя, в качестве данного адреса может использоваться любой IPv6-адрес маршрутизатора, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<IF> – имя интерфейса маршрутизатора, через который будут отправлены пакеты, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<TUN> – имя туннеля маршрутизатора, через который будут отправлены пакеты, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#);

<DSCP> – DSCP-приоритет в соответствии с RFC 2474, принимает значение [0..63], значение по умолчанию 0;

<PORT> – номер TCP/UDP-порта, принимает значение [1..65535], значение по умолчанию 53 для UDP и 80 для TCP;

<GW-ADDR> – IP-адрес шлюза, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. При указании данного параметра в исходящий пакет добавляется IP source routing опция, которая сообщает маршрутизатору, через какой шлюз должен маршрутизироваться пакет в сети. На большинство маршрутизаторов отключена маршрутизация по данной опции из соображений безопасности;

<GWIPV6-ADDR> – IPv6-адрес шлюза, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF]. При указании данного параметра в исходящий пакет добавляется IP source routing опция, которая сообщает маршрутизатору, через какой шлюз должен маршрутизироваться пакет в сети. На большинство маршрутизаторов отключена маршрутизация по данной опции из соображений безопасности.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример:

```
esr# traceroute 192.168.27.128
traceroute to 192.168.27.128 (192.168.27.128), 30 hops max, 60 byte packets
 1 192.168.16.1 (192.168.16.1)  1.240 ms  1.546 ms  1.883 ms
 2 192.168.27.128 (192.168.27.128)  0.451 ms  0.437 ms  0.411 ms
```

## uptime

Данной командой осуществляется просмотр продолжительности времени работы устройства.

### Синтаксис

```
uptime
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример:

```
esr# uptime
System uptime (d,h:m:s):      00,00:26:35
```

## verify

Данной командой выполняется расчет хеш-суммы для отдельного файла с использованием указанного алгоритма хеширования.

### Синтаксис

```
verify <ALGORITHM> <FILE>
```

### Параметры

<ALGORITHM> – алгоритм хеширования, принимает значения [md5, sha2-256, sha2-512].

<FILE> – путь и имя файла для проверки, может принимать следующие значения:

- usb://usb\_name:/PATH
- mmc://mmc\_name:/PATH (кроме esr-10)
- system:candidate-config
- system:running-config
- system:factory-config
- system:default-config
- system:firmware-image-1
- system:firmware-image-2
- system:boot-1
- system:boot-2
- flash:critlog/FILE
- flash:syslog/FILE
- flash:backup/FILE
- flash:data/FILE
- tmpsys:syslog/FILE

### Необходимый уровень привилегий

15

### Командный режим

ROOT

### Пример:

```
esr# verify md5 system:firmware-image-1
system:firmware-image-1 16ef38a292e96ce972e910da6db2d1f4
```

### verify filesystem

Данной командой запускается процесс расчета хеш-сумм для всех системных файлов маршрутизатора и сравнения с эталонными значениями. В результате выводится информация о соответствии рассчитанных хеш-сумм эталонным.

### Синтаксис

```
verify filesystem [detailed]
```

### Параметры

detailed – ключ, отвечающий за подробный вывод информации о проверке каждого файла.

### Необходимый уровень привилегий

15

### Командный режим

ROOT

**Пример:**

```

esr# verify filesystem
Total:          2949
Success:        2949
Changed:        0
Deleted:        0

Filesystem verification success.

```

**verify storage-device**

Данной командой запускается процесс проверки внешнего носителя на наличие ошибок.

**Синтаксис**

```
verify storage-device { usb-dev://<ID> | usb://<USB-NAME> | mmc }
```

**Параметры**

<ID> – номер подключенного USB-носителя по порядку;

<USB-NAME> – имя подключенного USB-носителя. Имя можно узнать в выводе команды [show storage-devices](#).

mmc – ключ для проверки установленного mmc-носителя.

**Необходимый уровень привилегий**

15

**Командный режим**

ROOT

**Пример:**

```

esr# verify storage-device usb://BC1E-2E16
Device verify
Do you really want to continue? (y/N): y
CP437: Invalid argument
fsck.fat 4.0 (2016-05-06)
/dev/sda2: 14 files, 63908/255496 clusters
Device verification success.

```

## 6 Управление программным обеспечением и конфигурацией

- archive
- auto
- banner exec
- banner login
- boot host auto-config
- boot host auto-update
- boot system
- by-commit
- clear storage-device
- commit
- commit check
- confirm
- copy
- count-backup
- delete
- dir
- merge
- path
- restore
- rollback
- save
- show bootvar
- show boot-licence
- show candidate-config
- show configuration changes
- show crypto certificates
- show licence
- show running-config
- show storage-devices
- show usb
- show version
- time-period
- type
- unmount storage-device
- update crypto default

### archive

Данной командой осуществляется переход в режим настройки параметров резервирования конфигурации.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию для параметров резервирования конфигурации.

### Синтаксис

```
[no] archive
```

### Параметры

Команда не содержит параметров.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример:**

```
esr(config)# archive
esr(config-archive)#
```

**auto**

Данной командой включается режим создания файла резервной конфигурации на удаленном сервере и/или локально через указанный промежуток времени (раздел [time-period](#)).

Использование отрицательной формы команды (no) выключает режим отправки через указанный промежуток времени.

**Синтаксис**

```
[no] auto
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-ARCHIVE

**Пример:**

```
esr(config-archive)# auto
```

**banner exec**

Данной командой задаётся текстовое сообщение, выводимое после успешной аутентификации пользователя на маршрутизаторе.

Использование отрицательной формы команды (no) удаляет текстовое сообщение, выводимое после успешной аутентификации пользователя на маршрутизаторе.



**Синтаксис**

```
banner exec <BANNER>
no banner exec
```

**Параметры**

<BANNER> – текстовое сообщение, задаётся в кавычках строкой до 2047 символов.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример:**

```
esr(config)# banner exec '                               ----- Hello! This is an Eltex router.-----
\n----- Unauthorized configuration changes are prosecuted by the law of the Russian Federation.
-----'
```

**banner login**

Данной командой задаётся текстовое сообщение, выводимое до аутентификации пользователя на маршрутизаторе.

**Синтаксис**

```
banner login <BANNER>
no banner login
```

**Параметры**

<BANNER> – текстовое сообщение, задаётся в кавычках строкой до 2047 символов.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример:**

```
esr(config)# banner login '                               ----- Hello! This is an Eltex router.-----
\n----- Unauthorized connection is prosecuted by the law of the Russian Federation. -----'
```

## boot host auto-config

Данная команда включает загрузку конфигурации маршрутизатора при помощи протокола DHCP Option 54 (IP-адрес TFTP-сервера) и 67 (имя файла конфигурации).

Использование отрицательной формы команды (no) отключает загрузку конфигурации маршрутизатора при помощи протокола DHCP Option 66 и 67.

### Синтаксис

```
[no] boot host auto-config
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

10

### Значение по умолчанию

Отключено.

### Командный режим

CONFIG

### Пример:

```
esr(config)# boot host auto-config
```

## boot host auto-update

Данная команда включает загрузку ПО маршрутизатора при помощи протокола DHCP Option 54 (IP-адрес TFTP-сервера) и имя файла ПО в поле FILE в DHCP OFFER. При использовании совместно с [boot host auto-config](#) обновление ПО имеет более высокий приоритет.

Использование отрицательной формы команды (no) отключает загрузку ПО маршрутизатора при помощи протокола DHCP Option 150 и 67.

### Синтаксис

```
[no] boot host auto-update
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

10

**Значение по умолчанию**

Отключено.

**Командный режим**

CONFIG

**Пример:**

```
esr(config)# boot host auto-update
```

**boot system**

Данная команда служит для выбора активного образа программного обеспечения, загруженного на устройство.

**Синтаксис**

```
boot system <IMAGE>
```

**Параметры**

<IMAGE> – название образа программного обеспечения, который будет загружаться на устройство:

- image-1 – следующая загрузка устройства будет выполнена из первого образа ПО;
- image-2 – следующая загрузка устройства будет выполнена из второго образа ПО;
- inactive – следующая загрузка устройства будет выполнена с неактивного образа ПО.

**Необходимый уровень привилегий**

15

**Командный режим**

ROOT

**Пример:**

```
esr# boot system image-2
```

**by-commit**

Данной командой включается режим отправки файла конфигурации на сервер резервирования после удачного применения конфигурации.

Использование отрицательной формы команды (no) выключает режим отправки после удачного применения конфигурации.

**Синтаксис**

```
[no] by-commit
```

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-ARCHIVE

## Пример:

```
esr(config-archive)# by-commit
```

## clear storage-device

Данная команда служит для форматирования всего устройства хранения данных или конкретной партии. Если жесткий диск без разделов, то для форматирования используется команда `clear storage-device hdd-dev://`.

## Синтаксис

```
clear storage-device { usb-dev://<SLOT_NUMBER> <NEW_PARTITION> | usb://<PARTITION>:/
<NEW_PARTITION> | mmc <NEW_PARTITION> | hdd-dev://<SLOT_NUMBER> <NEW_PARTITION> | hdd://
<PARTITION>:/ <NEW_PARTITION> }
```

## Параметры

<SLOT\_NUMBER> – номер слота, для WLC всегда 1;

<NEW\_PARTITION> – наименование новой партии, задается строкой от 1 до 15 символов;

<PARTITION> – наименование партии, которую требуется отформатировать, задается строкой от 1 до 15 символов.

## Необходимый уровень привилегий

15

## Командный режим

ROOT

## Пример:

```
wlc# clear storage-device hdd-dev://1 new_hdd
Formatting will erase all data on this device
Do you really want to continue? (y/N): y
mke2fs 1.43.1 (08-Jun-2016)
Device clear success.
wlc# 2024-09-16T07:04:18+00:00 %HDD-I-CHANGE: 'new_hdd' has been inserted!
```

## commit

Данная команда позволяет применить (сделать действующими) изменения конфигурации. RUNNING-конфигурация замещается конфигурацией CANDIDATE. Для того чтобы примененные изменения стали постоянно действующими, эту операцию необходимо подтвердить командой *confirm* в течение времени, не превышающего время действия таймера подтверждения (по умолчанию 600 секунд, изменяется при помощи команды [system config-confirm timeout](#)).

### Синтаксис

```
commit [ comment <COMMENT> ] [ confirm-timeout <TIME> ]
```

### Параметры

<COMMENT> – комментарий к данному изменению рабочей конфигурации. Задаётся текстовой строкой до 31 символа;

<TIMEOUT> – интервал времени в секундах после ввода команды *commit* до восстановления предыдущей версии конфигурации при отсутствии команды *confirm*, принимает значение в секундах [120..86400].

### Значение по умолчанию

<COMMENT> – пустой;

<TIMEOUT> – наследуется значение, установленное командой [system config-confirm timeout](#).

### Необходимый уровень привилегий

10

### Командный режим

ROOT

CHANGE-EXPIRED-PASSWORD

### Пример:

```
esr# commit
```

Изменения конфигурации применены.

## commit check

Данная команда позволяет провалидировать конфигурацию CANDIDATE, пока изменения не применены.

### Синтаксис

```
commit check
```

### Параметры

Отсутствуют.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример:

```
esr# commit check
```

Изменения конфигурации применены.

### confirm

Команда предназначена для подтверждения применения конфигурации. Если в течение заданного времени (по умолчанию 600 секунд, изменяется при помощи команды [system config-confirm timeout](#)) после применения конфигурации командой *commit* не было введено подтверждение, произойдет автоматический откат на действующую ранее конфигурацию. Автоматическая система откатов предотвращает ситуации потери связи с устройством.

### Синтаксис

```
confirm
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

Значение по умолчанию  
110

### Командный режим

ROOT

CHANGE-EXPIRED-PASSWORD

### Пример:

```
esr# confirm
```

Подтверждение изменений в конфигурации.

## copy

Данная команда служит для копирования файлов между различными источниками и получателями.

### Синтаксис

copy <SOURCE> <DESTINATION>

### Параметры

<SOURCE> – источник, задаётся в виде:

- tftp://<ip>[<port>]:/<path> – адрес файла на TFTP-сервере, где:
  - <ip> – IP-адрес TFTP-сервера;
  - <port> – порт, который слушает TFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный путь к файлу на TFTP-сервере.
- tftp://<ipv6>[%<interface>][<port>]:/<path> – адрес файла на TFTP-сервере, где:
  - <ipv6> – IPv6-адрес TFTP-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <port> – порт, который слушает TFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный путь к файлу на TFTP-сервере.
- ftp://[<user>[:<password>]@]<ip>[<port>]:/<path>
  - <ip> – IP-адрес FTP-сервера;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip ftp client username*, описанной в [ip ftp client username](#));
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip ftp client password*, описанной в [ip ftp client password](#));
  - <port> – порт, который слушает TFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный или абсолютный путь к файлу на FTP-сервере.
- ftp://[<user>[:<password>]@]<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – IPv6-адрес FTP-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip ftp client username*, описанной в [ip ftp client username](#));
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip ftp client password*, описанной в [ip ftp client password](#));
  - <port> – порт, который слушает TFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный или абсолютный путь к файлу на FTP-сервере.
- sftp://[<user>[:<password>]@]<ip>[<port>]:/<path>
  - <ip> – IP-адрес SFTP-сервера;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip sftp client username*, описанной в [ip sftp client username](#));
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip sftp client password*, описанной в [ip sftp client password](#));
  - <port> – порт, который слушает SFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный путь к файлу на SFTP-сервере.
- sftp://[<user>[:<password>]@]<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – IPv6-адрес SFTP-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip sftp client username*, описанной в [ip sftp client username](#));
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip sftp client password*, описанной в [ip sftp client password](#));

- <port> – порт, который слушает SFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
- <path> – относительный путь к файлу на SFTP-сервере.
- scp://[<user>:<password>@]<ip>[<port>]:/<path>
  - <ip> – IP-адрес сервера;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip ssh client username*, описанной в [ip ssh client username](#));
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip ssh client password*, описанной в [ip ssh client password](#));
  - <port> – порт, который слушает TFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный или абсолютный путь к файлу на сервере.
- scp://[<user>:<password>@]<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – IPv6-адрес сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip ssh client username*, описанной в [ip ssh client username](#));
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip ssh client password*, описанной в [ip ssh client password](#));
  - <port> – порт, который слушает TFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный или абсолютный путь к файлу на сервере.
- http://<ip>[<port>]:/<path> – адрес файла на HTTP-сервере, где:
  - <ip> – IP-адрес HTTP-сервера;
  - <port> – порт, на котором запущен HTTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный путь к файлу на HTTP-сервере.
- http://<ipv6>[%<interface>][<port>]:/<path> – адрес файла на HTTP-сервере, где:
  - <ipv6> – IPv6-адрес HTTP-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <port> – порт, на котором запущен HTTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный путь к файлу на TFTP-сервере.
- https://[ca/<CA-FILE>]:server-crt/<SC-FILE>[:server-key/<SK-FILE>]@<ip>[<port>]:/<path> – адрес файла на HTTPS-сервере, где:
  - <CA-FILE> – имя файла сертификата удостоверяющего сервера в соответствующем разделе памяти маршрутизатора;
  - <SC-FILE> – имя файла публичного сертификата сервера в соответствующем разделе памяти маршрутизатора;
  - <SK-FILE> – имя файла приватного ключа сервера в соответствующем разделе памяти маршрутизатора;
  - <ip> – IP-адрес HTTP-сервера;
  - <port> – порт, на котором запущен HTTPS-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный путь к файлу на HTTP-сервере.
- https://[ca/<CA-FILE>]:server-crt/<SC-FILE>[:server-key/<SK-FILE>]@<ipv6>[%<interface>][<port>]:/<path> – адрес файла на HTTPS-сервере, где:
  - <CA-FILE> – имя файла сертификата удостоверяющего сервера в соответствующем разделе памяти маршрутизатора;
  - <SC-FILE> – имя файла публичного сертификата сервера в соответствующем разделе памяти маршрутизатора;
  - <SK-FILE> – имя файла приватного ключа сервера в соответствующем разделе памяти маршрутизатора;
  - <ipv6> – IPv6-адрес HTTPS-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <port> – порт, на котором запущен HTTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);



- `<path>` – относительный путь к файлу на TFTP-сервере.
- `usb://usb_name:/PATH`
  - `usb_name` – имя, назначенное USB-носителю. Возможно посмотреть командой `show storage-devices` (см. раздел [show storage-devices](#));
  - `<path>` – путь к файлу на USB-носителе.
- `mmc://mmc_name:/PATH` (кроме ESR-10/12V/12VF/15)
  - `mmc_name` – имя, назначенное MMC-носителю. Возможно посмотреть командой `show storage-devices` (см. раздел [show storage-devices](#));
  - `<path>` – путь к файлу на MMC-носителе.
- `system:factory-config` – заводская конфигурация;
- `system:default-config` – конфигурация по умолчанию (пустая);
- `system:running-config` – текущая конфигурация;
- `system:candidate-config` – конфигурация, которая будет применена после выполнения команды `commit`;
- `system:firmware` – программное обеспечение устройства. Копирование производится с неактивного образа программного обеспечения устройства;
- `system:boot-1` – первичный загрузчик устройства (SBI, bl1, x-loader);
- `system:boot-2` – вторичный загрузчик устройства (U-Boot, boot) на всех устройствах, за исключением ESR-15/30/31;
- `system:boot` – вторичный загрузчик устройства (U-Boot) на ESR-15/30/31;
- `system:licence` – функциональная лицензия устройства;
- `flash:critlog/FILE` – папка для сохранения сообщений ядра за все время работы устройства;
- `flash:syslog/FILE` – папка для сохранения логов текущей сессии, сохраняется после перезагрузки;
- `tmpsys:syslog/FILE` – папка для сохранения логов текущей сессии, не сохраняется после перезагрузки;
- `flash:backup/FILE` – папка для сохранения резервных копий текущих конфигураций маршрутизатора;
- `flash:data/FILE` – папка для скачивания файлов с маршрутизатора;
- `crypto:cert/FILE` – папка для хранения сертификатов;
- `crypto:crl/FILE` – папка для хранения списков отозванных сертификатов удостоверяющих центров;
- `crypto:private-key/FILE` – папка для хранения приватных ключей;
- `crypto:public-key/FILE` – папка для хранения публичных ключей;
- `crypto:dh/FILE` – папка для хранения ключей Диффи-Хеллмана;
- `crypto:px/FILE` – папка для хранения контейнеров PKCS#12;
- `crypto:csr/FILE` – папка для хранения запросов на сертификацию X.509;
- `crypto:ta/FILE` – папка для хранения ТА-ключей OpenVPN.

`<DESTINATION>` – назначение, задаётся в виде:

- `tftp://<ip>[<port>]:/<path>` – адрес файла на TFTP-сервере, где:
  - `<ip>` – IP-адрес TFTP-сервера;
  - `<port>` – порт, который слушает TFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - `<path>` – относительный путь к файлу на TFTP-сервере.
- `tftp://<ipv6>[%<interface>][<port>]:/<path>` – адрес файла на TFTP-сервере, где:
  - `<ipv6>` – IPv6-адрес TFTP-сервера;
  - `<interface>` – исходящий сетевой интерфейс для link-local адресов;
  - `<port>` – порт, который слушает TFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - `<path>` – относительный путь к файлу на TFTP-сервере.
- `ftp://[<user>[:<password>]@]<ip>[<port>]:/<path>`
  - `<ip>` – IP-адрес FTP-сервера;
  - `<user>` – имя пользователя (настроить имя пользователя по умолчанию можно командой `ip ftp client username`, описанной в [ip ftp client username](#));
  - `<password>` – пароль (настроить пароль по умолчанию можно командой `ip ftp client password`, описанной в [ip ftp client password](#));
  - `<port>` – порт, который слушает TFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - `<path>` – относительный или абсолютный путь к файлу на FTP-сервере.

- ftp://[<user>[:<password>]@]<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – IPv6-адрес FTP-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip ftp client username*, описанной в [ip ftp client username](#));
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip ftp client password*, описанной в [ip ftp client password](#));
  - <port> – порт, который слушает TFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный или абсолютный путь к файлу на FTP-сервере.
- sftp://[<user>[:<password>]@]<ip>[<port>]:/<path>
  - <ip> – IP-адрес SFTP-сервера;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip sftp client username*, описанной в [ip sftp client username](#));
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip sftp client password*, описанной в [ip sftp client password](#));
  - <port> – порт, который слушает SFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный путь к файлу на SFTP-сервере.
- sftp://[<user>[:<password>]@]<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – IPv6-адрес SFTP-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip sftp client username*, описанной в [ip sftp client username](#));
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip sftp client password*, описанной в [ip sftp client password](#));
  - <port> – порт, который слушает SFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный путь к файлу на SFTP-сервере.
- scp://[<user>[:<password>]@]<ip>[<port>]:/<path>
  - <ip> – IP-адрес сервера;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip ssh client username*, описанной в [ip ssh client username](#));
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip ssh client password*, описанной в [ip ssh client password](#));
  - <port> – порт, который слушает TFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный или абсолютный путь к файлу на сервере.
- scp://[<user>[:<password>]@]<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – IPv6-адрес сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip ssh client username*, описанной в [ip ssh client username](#));
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip ssh client password*, описанной в [ip ssh client password](#));
  - <port> – порт, который слушает TFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный или абсолютный путь к файлу на сервере.
- http://<ip>[<port>]:/<path> – адрес файла на HTTP-сервере, где:
  - <ip> – IP-адрес HTTP-сервера;
  - <port> – порт, на котором запущен HTTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный путь к файлу на HTTP-сервере.
- http://[<ipv6>[%<interface>][<port>]:/<path> – адрес файла на HTTP-сервере, где:
  - <ipv6> – IPv6-адрес HTTP-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <port> – порт, на котором запущен HTTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);

- <path> – относительный путь к файлу на TFTP-сервере.
- https://[ca/<CA-FILE>]:server-crt/<SC-FILE>]:server-key/<SK-FILE>]@<ip>[<port>]:/<path> – адрес файла на HTTPS-сервере, где:
  - <CA-FILE> – имя файла сертификата удостоверяющего сервера в соответствующем разделе памяти маршрутизатора;
  - <SC-FILE> – имя файла публичного сертификата сервера в соответствующем разделе памяти маршрутизатора;
  - <SK-FILE> – имя файла приватного ключа сервера в соответствующем разделе памяти маршрутизатора;
  - <ip> – IP-адрес HTTP-сервера;
  - <port> – порт, на котором запущен HTTPS-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный путь к файлу на HTTP-сервере.
- https://[ca/<CA-FILE>]:server-crt/<SC-FILE>]:server-key/<SK-FILE>]@<ipv6>[%<interface>][<port>]:/<path> – адрес файла на HTTPS-сервере, где:
  - <CA-FILE> – имя файла сертификата удостоверяющего сервера в соответствующем разделе памяти маршрутизатора;
  - <SC-FILE> – имя файла публичного сертификата сервера в соответствующем разделе памяти маршрутизатора;
  - <SK-FILE> – имя файла приватного ключа сервера в соответствующем разделе памяти маршрутизатора;
  - <ipv6> – IPv6-адрес HTTPS-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <port> – порт, на котором запущен HTTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный путь к файлу на TFTP-сервере.
- usb://usb\_name:/PATH
  - usb\_name – имя, назначенное USB-носителю. Возможно посмотреть командой *show storage-devices* (см. раздел [show storage-devices](#));
  - <path> – путь к файлу на USB-носителе.
- mmc://mmc\_name:/PATH (кроме ESR-10/12V/12VF/15/15R/15VF):
  - mmc\_name – имя, назначенное MMC-носителю. Возможно посмотреть командой *show storage-devices* (см. раздел [show storage-devices](#));
  - <path> – путь к файлу на MMC-носителе.
- system:candidate-config – конфигурация, которая будет применена после выполнения команды *commit*;
- system:firmware – программное обеспечение устройства. Копирование производится с неактивного образа программного обеспечения устройства;
- system:boot-2 – вторичный загрузчик устройства (U-Boot, boot) на всех устройствах, за исключением ESR-15/30/31;
- system:boot – вторичный загрузчик устройства (U-Boot) на ESR-15/30/31;
- system:licence – функциональная лицензия устройства;
- system:boot-licence – загрузочная лицензия устройства;
- system:cluster-unit-licences – папка для хранения лицензий устройств в кластере;
- flash:data/FILE – папка для скачивания файлов с маршрутизатора;
- crypto:cert/FILE – папка для хранения сертификатов;
- crypto:crl/FILE – папка для хранения списков отозванных сертификатов удостоверяющих центров;
- crypto:private-key/FILE – папка для хранения приватных ключей;
- crypto:public-key/FILE – папка для хранения публичных ключей;
- crypto:dh/FILE – папка для хранения ключей Диффи-Хеллмана;
- crypto:pfx/FILE – папка для хранения контейнеров PKCS#12;
- crypto:csr/FILE – папка для хранения запросов на сертификацию X.509;
- crypto:ta/FILE – папка для хранения ТА-ключей OpenVPN.

**Необходимый уровень привилегий**

15

**Командный режим**

ROOT

**Пример 1**

```
esr# copy tftp://10.100.100.1/esr.cfg system:candidate-config
```

**Пример 2**

```
esr# copy tftp://10.100.100.1/crl.pem crypto:crl/crl.pem
```

**count-backup**

Данной командой устанавливается максимальное количество локально сохраняемых резервных копий конфигураций.

При использовании отрицательной формы команды (no) устанавливается значение по умолчанию.

**Синтаксис**

```
count-backup <NUM>
```

```
no count-backup
```

**Параметры**

<NUM> – максимальное количество локально сохраняемых резервных копий конфигураций. Принимает значения в диапазоне [1..100].

**Значение по умолчанию**

1

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-ARCHIVE

**Пример:**

```
esr(config-archive)# count-backup 20
```

## delete

Данная команда служит для удаления лицензий, сертификатов, ключей.

### Синтаксис

```
delete <FILE>
```

### Параметры

<FILE> – тип файла, может принимать следующие значения (при удалении из папки необходимо указать название файла):

- flash:critlog/FILE – папка для сохранения сообщений ядра за все время работы устройства;
- flash:syslog/FILE – папка для сохранения логов текущей сессии, сохраняется после перезагрузки;
- flash:backup/FILE – папка для сохранения резервных копий текущих конфигураций маршрутизатора;
- flash:data/FILE – папка для скачивания файлов с маршрутизатора;
- tmpsys:syslog/FILE – папка для сохранения логов текущей сессии, не сохраняется после перезагрузки;
- system:access-points-firmwares/[FILE] – папка для хранения ПО точек доступа.
- system:licence – лицензия для активации функционала, требующего лицензию;
- system:cluster-unit-licences/[Serial number] – папка для хранения лицензий устройств в кластере;
- crypto:cert/FILE – папка для хранения сертификатов;
- crypto:dh/FILE – папка для хранения ключей Диффи-Хеллмана;
- crypto:ta/FILE – папка для хранения ТА-ключей OpenVPN;
- crypto:crl/FILE – папка для хранения списков, отозванных сертификатов удостоверяющих центров;
- crypto:csr/FILE – папка для хранения запросов на сертификацию X.509;
- crypto:pfx/FILE – папка для хранения контейнеров PKCS#12;
- crypto:private-key/FILE – папка для хранения приватных ключей;
- crypto:public-key/FILE – папка для хранения публичных ключей;
- usb://usb\_name:/FILE:
  - usb\_name – имя, назначенное USB-носителю. Возможно посмотреть командой *show storage-devices* (см. раздел [show storage-devices](#));
  - <FILE> – путь к файлу и его имя на USB-носителе.
- hdd://hdd\_name:/PATH ( для WLC-15, WLC-30, WLC-3200)
  - hdd\_name – имя назначенное HDD\SSD-носителю. Возможно посмотреть командой *show storage-devices hdd* (см. раздел [show storage-devices](#));
  - <PATH> – путь к файлу на HDD\SSD-носителе.
- mmc://mmc\_name:/<FILE> (для WLC-30, WLC-3200, ESR-3200)
  - mmc\_name – имя, назначенное MMC-носителю. Возможно посмотреть командой *show storage-devices* (см. раздел [show storage-devices](#));
  - <FILE> – путь к файлу и его имя на MMC-носителе.

### Необходимый уровень привилегий

15

### Командный режим

ROOT

**Пример:**

```
esr# delete crypto:dh/dh.key
```

**dir**

Данная команда позволяет отобразить содержимое внешних носителей информации (USB/MMC/HDD-носителей) и локальных разделов маршрутизатора.

**Синтаксис**

```
dir <PATH>
{ flash:critlog/FILE | flash:syslog/FILE | flash:backup/FILE | flash:data/FILE |
tmpsys:syslog/FILE | system:access-points-firmwares/[FILE] | crypto:cert/FILE |
crypto:dh/FILE | crypto:ta/FILE | crypto:crl/FILE | crypto:csr/FILE | crypto:px/FILE |
crypto:private-key/FILE | crypto:public-key/FILE | usb://<USB-device-name>/<PATH> |
mmc://<MMC-device-name>/<PATH> | hdd://<HDD-device-name>/<PATH> }
```

**Параметры**

<PATH> – имя локального раздела или путь до папки на внешнем носителе. Принимает значения:

- flash:critlog/FILE – папка для сохранения сообщений ядра за все время работы устройства;
- flash:syslog/FILE – папка для сохранения логов текущей сессии, сохраняется после перезагрузки;
- flash:backup/FILE – папка для сохранения резервных копий текущих конфигураций маршрутизатора;
- flash:data/FILE – папка для скачивания файлов с маршрутизатора;
- tmpsys:syslog/FILE – папка для сохранения логов текущей сессии, не сохраняется после перезагрузки;
- system:access-points-firmwares/[FILE] – папка для хранения ПО точек доступа.
- crypto:cert/FILE – папка для хранения сертификатов;
- crypto:dh/FILE – папка для хранения ключей Диффи-Хеллмана;
- crypto:ta/FILE – папка для хранения ТА-ключей OpenVPN;
- crypto:crl/FILE – папка для хранения списков отозванных сертификатов удостоверяющих центров;
- crypto:csr/FILE – папка для хранения запросов на сертификацию X.509;
- crypto:px/FILE – папка для хранения контейнеров PKCS#12;
- crypto:private-key/FILE – папка для хранения приватных ключей;
- crypto:public-key/FILE – папка для хранения публичных ключей;
- usb://usb\_name:/PATH
  - usb\_name – имя, назначенное USB-носителю. Возможно посмотреть командой *show storage-devices usb* (см. раздел [show storage-devices](#));
  - <PATH> – путь к файлу на USB-носителе.
- hdd://hdd\_name:/PATH ( для WLC-15, WLC-30, WLC-3200)
  - hdd\_name – имя назначенное HDD\SSD-носителю. Возможно посмотреть командой *show storage-devices hdd* (см. раздел [show storage-devices](#))
  - <PATH> – путь к файлу на HDD\SSD-носителе.
- mmc://mmc\_name:/PATH (для WLC-30, WLC-3200, ESR-3200)
  - mmc\_name – имя, назначенное MMC-носителю. Возможно посмотреть командой *show storage-devices mmc* (см. раздел [show storage-devices](#));
  - <PATH> – путь к файлу на MMC-носителе.

**Необходимый уровень привилегий**

1



## Командный режим

ROOT

### Пример:

```

esr# dir mmc://EF28-D074
Name                                     Type      Size      --
-----
esr1000-1.4.0-build21.uboot             File      0.00      B
.Trash-1000                              Directory 0.00      B

```

## merge

Данная команда служит для объединения локального или загружаемого файла конфигурации с candidate-config.

### Синтаксис

```
merge <SOURCE> system:candidate-config
```

### Параметры

<SOURCE> – источник, задаётся в виде:

- tftp://<ip>[<port>]:/<path> – адрес файла на TFTP-сервере, где:
  - <ip> – IP-адрес TFTP-сервера;
  - <port> – порт, который слушает TFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный путь к файлу на TFTP-сервере.
- tftp://<ipv6>[%<interface>][<port>]:/<path> – адрес файла на TFTP-сервере, где:
  - <ipv6> – IPv6-адрес TFTP-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <port> – порт, который слушает TFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный путь к файлу на TFTP-сервере.
- ftp://[<user>[:<password>]@]<ip>[<port>]:/<path>
  - <ip> – IP-адрес FTP-сервера;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip ftp client username*, описанной в [ip ftp client username](#));
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip ftp client password*, описанной в [ip ftp client password](#));
  - <port> – порт, который слушает TFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный или абсолютный путь к файлу на FTP-сервере.
- ftp://[<user>[:<password>]@]<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – IPv6-адрес FTP-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip ftp client username*, описанной в [ip ftp client username](#));
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip ftp client password*, описанной в [ip ftp client username](#));
  - <port> – порт, который слушает TFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный или абсолютный путь к файлу на FTP-сервере.
- sftp://[<user>[:<password>]@]<ip>[<port>]:/<path>

- <ip> – IP-адрес SFTP-сервера;
- <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip sftp client username*, описанной в [ip sftp client username](#));
- <password> – пароль (настроить пароль по умолчанию можно командой *ip sftp client password*, описанной в [ip sftp client password](#));
- <port> – порт, который слушает SFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
- <path> – относительный путь к файлу на SFTP-сервере.
- sftp://<user>[:<password>]@<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – IPv6-адрес SFTP-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip sftp client username*, описанной в [ip sftp client username](#));
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip sftp client password*, описанной в [ip sftp client password](#));
  - <port> – порт, который слушает SFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный путь к файлу на SFTP-сервере.
- scp://<user>:<password>@<ip>[<port>]:/<path>
  - <ip> – IP-адрес сервера;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip ssh client username*, описанной в [ip ssh client username](#));
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip ssh client password*, описанной в [ip ssh client password](#));
  - <port> – порт, который слушает TFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный или абсолютный путь к файлу на сервере.
- scp://<user>:<password>@<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – IPv6-адрес сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip ssh client username*, описанной в [ip ssh client username](#));
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip ssh client password*, описанной в [ip ssh client password](#));
  - <port> – порт, который слушает TFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный или абсолютный путь к файлу на сервере.
- http://<ip>[<port>]:/<path> – адрес файла на HTTP-сервере, где:
  - <ip> – IP-адрес HTTP-сервера;
  - <port> – порт, на котором запущен HTTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный путь к файлу на HTTP-сервере.
- http://<ipv6>[%<interface>][<port>]:/<path> – адрес файла на HTTP-сервере, где:
  - <ipv6> – IPv6-адрес HTTP-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <port> – порт, на котором запущен HTTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный путь к файлу на TFTP-сервере.
- https://[ca/<CA-FILE>]:server-crt/<SC-FILE>[:server-key/<SK-FILE>]@<ip>[<port>]:/<path> – адрес файла на HTTPS-сервере, где:
  - <CA-FILE> – имя файла сертификата удостоверяющего сервера в соответствующем разделе памяти маршрутизатора;
  - <SC-FILE> – имя файла публичного сертификата сервера в соответствующем разделе памяти маршрутизатора;
  - <SK-FILE> – имя файла приватного ключа сервера в соответствующем разделе памяти маршрутизатора;
  - <ip> – IP-адрес HTTP-сервера;



- <port> – порт, на котором запущен HTTPS-сервер, отделяется от IP-адреса символом «#» или «:»;
- <path> – относительный путь к файлу на HTTP-сервере.
- https://[ca/<CA-FILE>]:server-crt/<SC-FILE>[:server-key/<SK-FILE>]@<ipv6>[%<interface>][<port>]:/<path> – адрес файла на HTTPS-сервере, где:
  - <CA-FILE> – имя файла сертификата удостоверяющего сервера в соответствующем разделе памяти маршрутизатора;
  - <SC-FILE> – имя файла публичного сертификата сервера в соответствующем разделе памяти маршрутизатора;
  - <SK-FILE> – имя файла приватного ключа сервера в соответствующем разделе памяти маршрутизатора;
  - <ipv6> – IPv6-адрес HTTPS-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <port> – порт, на котором запущен HTTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный путь к файлу на TFTP-сервере.
- flash:data/FILE – папка для скачивания файлов с маршрутизатора.

## Необходимый уровень привилегий

15

## Командный режим

ROOT

Пример

```
esr# merge tftp://10.100.100.1/esr.cfg system:candidate-config
```

## path

Данной командой определяются протокол, адрес сервера, а также расположение и префикс имени файла на сервере. При выполнении резервирования к префиксу имени файла добавляется текущее время и дата в формате ГГГГММДД\_ЧЧММСС.

Использование отрицательной формы команды (no) удаляет установленное значение.

## Синтаксис

path <PATH>

no path

## Параметры

<PATH> – формат пути до папки на удаленном сервере по протоколам tftp/ftp/sftp/scp в одном из следующих форматов:

- tftp://<ip>[<port>]:/<path> – адрес файла на TFTP-сервере, где:
  - <ip> – IP-адрес TFTP-сервера;
  - <port> – порт, который слушает TFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный путь к файлу на TFTP-сервере.
- tftp://<ipv6>[%<interface>][<port>]:/<path> – адрес файла на TFTP-сервере, где:
  - <ipv6> – IPv6-адрес TFTP-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;

- <port> – порт, который слушает TFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
- <path> – относительный путь к файлу на TFTP-сервере.
- ftp://[<user>:<password>]@<ip>[<port>]:/<path>
  - <ip> – IP-адрес FTP-сервера;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip ftp client username*, описанной в [ip ftp client username](#));
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip ftp client password*, описанной в [ip ftp client password](#));
  - <port> – порт, который слушает TFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный или абсолютный путь к файлу на FTP-сервере.
- ftp://[<user>:<password>]@[<ipv6>[%<interface>]][<port>]:/<path>
  - <ipv6> – IPv6-адрес FTP-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip ftp client username*, описанной в [ip ftp client username](#));
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip ftp client password*, описанной в [ip ftp client username](#));
  - <port> – порт, который слушает TFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный или абсолютный путь к файлу на FTP-сервере.
- sftp://[<user>:<password>]@<ip>[<port>]:/<path>
  - <ip> – IP-адрес SFTP-сервера;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip sftp client username*, описанной в [ip sftp client username](#));
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip sftp client password*, описанной в [ip sftp client password](#));
  - <port> – порт, который слушает SFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный путь к файлу на SFTP-сервере.
- sftp://[<user>:<password>]@[<ipv6>[%<interface>]][<port>]:/<path>
  - <ipv6> – IPv6-адрес SFTP-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip sftp client username*, описанной в [ip sftp client username](#));
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip sftp client password*, описанной в [ip sftp client password](#));
  - <port> – порт, который слушает SFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный путь к файлу на SFTP-сервере.
- scp://[<user>:<password>]@<ip>[<port>]:/<path>
  - <ip> – IP-адрес сервера;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip ssh client username*, описанной в [ip ssh client username](#));
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip ssh client password*, описанной в [ip ssh client password](#));
  - <port> – порт, который слушает TFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный или абсолютный путь к файлу на сервере.
- scp://[<user>:<password>]@[<ipv6>[%<interface>]][<port>]:/<path>
  - <ipv6> – IPv6-адрес сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip ssh client username*, описанной в [ip ssh client username](#));
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip ssh client password*, описанной в [ip ssh client password](#));
  - <port> – порт, который слушает TFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);

- <path> – относительный или абсолютный путь к файлу на сервере.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-ARCHIVE

### Пример:

```
esr(config-archive)# path tftp://10.10.10.1:/esr-1000/config
```

### restore

Данная команда позволяет отменить примененную, но неподтвержденную конфигурацию и вернуться к последней подтвержденной. Команда применяется ко всей конфигурации устройства. Отмена изменений может быть выполнена только до ввода команды *confirm*. При выполнении команды *restore* происходит потеря неподтвержденной конфигурации.

### Синтаксис

restore

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример:

```
esr# restore
```

Осуществлен возврат к последней подтвержденной конфигурации.

### rollback

Данная команда позволяет изменить содержимое candidate-config.

Команда применяется ко всей конфигурации устройства.

### Синтаксис

```
rollback [to-backup <FILENAME>]
```

## Параметры

<FILENAME> – имя файла ранее сохраненной конфигурации из раздела flash:backup/ маршрутизатора.

## Значение по умолчанию

Без использования ключа "to-backup" в candidate-config копируется текущая "running-config".

## Необходимый уровень привилегий

10

## Командный режим

ROOT

## Пример:

```
esr# rollback
```

Произведена отмена всех непримененных изменений в конфигурации.

## save

Команда служит для сохранения CANDIDATE-конфигурации в постоянную память устройства.

## Синтаксис

save

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

10

## Командный режим

ROOT

## Пример:

```
esr# save
```

Сохранение текущей конфигурации на Flash-память устройства.

## show bootvar

Данная команда служит для просмотра информации об образах программного обеспечения, загруженных на устройство.

**Синтаксис**

```
show bootvar
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример:**

```
esr# show bootvar
Image      Version                               Date                               Status      After reboot
-----
1          1.24.0 build 1[5cd22b8]             2024-08-01 18:00:47             Not Active
2          1.24.0 build 1[5cd22b8]             2024-08-01 18:00:47             Active      *
```

**show boot-licence**

Данная команда служит для просмотра информации об активной лицензии устройства применяемых на этапе загрузки вторичного загрузчика.

**Синтаксис**

```
show boot-licence
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

5

**Командный режим**

ROOT

**Пример:**

```
esr# show boot-licence
Licence information
-----
Name:      Eltex
Version:   1.0
Type:      ESR-1000
```

```
S/N:      NP01000530
MAC:     A8:F9:4B:AA:44:BB
Features:
KSS - Kaspersky Security System
```

## show candidate-config

Данной командой осуществляется просмотр конфигурации устройства, которая будет установлена после применения настроек (команда *commit*).

### Синтаксис

```
show candidate-config [ full ] [ <SECTION> ]
```

### Параметры


*full* – ключ для отображения в конфигурации значений всех параметров, в том числе и ненастроенных.

<SECTION> – раздел конфигурации:

- *aaa* – настройка параметров аутентификации, авторизации и учета;
- *access-list* – конфигурация списков доступа;
- *bridges* – конфигурация сетевых мостов;
  - [ <NUM> ] – номер сетевого моста.
- *channel-group* – конфигурация группы агрегации каналов;
- *clock* – конфигурация системных часов маршрутизатора и NTP-протокола;
- *dhcp* – конфигурация DHCP-сервера, клиента и Relay-агента;
- *domain* – конфигурация domain lookup;
- *content-provider* – конфигурация источника обновлений правил распространяемых по коммерческой лицензии;
- *dual-homing* – конфигурация сервиса Dual Homing<sup>1</sup>;
- *extended* – расширенный вывод конфигурации;
- *failovers* – конфигурация резервирования;
- *hostname* – сетевое имя маршрутизатора;
- *interfaces* [<IF>] – конфигурация интерфейсов:
  - <IF> – наименование интерфейса, задается в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).
- *ip-address* – конфигурация IP-интерфейсов;
- *ipv6* [<SUBSECTION>] – конфигурация IPv6:
  - *address* – конфигурация IPv6-интерфейсов;
  - *dhcp* [<SUBSECTION>] – конфигурация IPv6 DHCP-сервисов:
  - *client* – конфигурация IPv6 DHCP-сервера;
    - *relay* – конфигурация IPv6 DHCP Relay-агента;
    - *server* – конфигурация IPv6 DHCP-клиента.
  - *routing* [<SUBSECTION>] – конфигурация IPv6-маршрутизации:
    - *bfd* – конфигурация протокола IPv6 BFD;
    - *bgp* – конфигурация протокола IPv6 BGP;
    - *ospf* – конфигурация протокола OSPFv3;
    - *prefix-lists* – конфигурация IPv6 префикс-листов;
    - *rip* – конфигурация протокола RIP;
    - *static* – конфигурация статических маршрутов.
  - *vrrp* – конфигурация IPv6 VRRP-протокола.
- *dialplan* – конфигурация плана нумерации;
- *lldp* – конфигурация протокола LLDP;
- *mac-address-table* – конфигурация таблицы MAC-адресов<sup>1</sup>;
- *mailservers* – конфигурация почтовых серверов и доменов;
- *mdns* – функционала mDNS;

- mirroring – конфигурация зеркалирования<sup>1</sup>;
- mpls – конфигурация протоколов технологии MPLS;
- mdns – конфигурация функционала mDNS;
- multiwan – конфигурация сервиса резервирования и балансировки WAN-интерфейсов;
- nat [<SUBSECTION>] – конфигурация сервиса NAT:
  - source – конфигурация сервиса Source NAT;
  - destination – конфигурация сервиса Destination NAT;
  - alg – конфигурация алгоритма NAT.
- netflow – конфигурация Netflow-протокола;
- object-groups [ <TYPE> [ <NAME> ] ] – конфигурация профилей;
  - <TYPE> – тип профиля, принимает значения:
    - content-filter;
    - email;
    - network;
    - address-port;
    - application;
    - mac;
    - service;
    - url.
  - <NAME> – имя профиля заданного типа.
- port-security – конфигурация Port Security<sup>1</sup>;
- qos – конфигурация QoS;
- remote-access [<SUBSECTION>] – конфигурация профилей L2TP over IPsec и PPTP-серверов;
  - l2tp – конфигурация профилей L2TP over IPsec-серверов;
  - pptp – конфигурация профилей PPTP-серверов;
  - openvpn – конфигурация профилей OpenVPN-серверов.
- remote-client – конфигурация удаленного доступа (SSH, Telnet, etc.);
- rmon – конфигурация RMON;
- routing [<SUBSECTION>] – конфигурация маршрутизации:
  - bfd – конфигурация протокола BFD;
  - bgp – конфигурация протокола BGP;
  - isis – конфигурация протокола IS-IS;
  - key-chains – конфигурация ключей аутентификации;
  - ospf – конфигурация протокола OSPF;
  - prefix-list – конфигурация префикс-листов;
  - rip – конфигурация протокола RIP;
  - route-maps – конфигурация маршрутных карт:
    - [ <RM-NAME> ] – имя маршрутной карты.
  - static – конфигурация статических маршрутов.
- security [<SUBSECTION>] – конфигурация сервисов IPsec VPN и Firewall:
  - ike – конфигурация IKE;
  - ipsec – конфигурация IPsec;
  - zone – конфигурация зон Firewall;
  - zone-pair – конфигурация переходов между зонами Firewall.
- sip – конфигурация протокола SIP;
  - profile – конфигурация SIP-профилей;
  - service – конфигурация SIP.
- sflow – конфигурация sFlow-протокола;
- snmp – конфигурация SNMP-сервера;
- spanning-tree – конфигурация протоколов семейства Spanning Tree;
- sla – конфигурация сервиса IP SLA;
- system – конфигурация общесистемных параметров;
- syslog – конфигурация сервиса Syslog;
- tracks – конфигурация Tracking-объектов.
- tunnels [<TYPE> [<NUM> ] ] – конфигурация туннелей:
  - <TYPE> – тип туннеля, принимает значения:

- gre – конфигурация GRE-туннелей;
- ip4ip4 – конфигурация IPv4 over IPv4-туннелей;
- l2tp – конфигурация L2TP-туннелей;
- l2tpv3 – конфигурация L2TPv3-туннелей;
- lt – конфигурация логических туннелей;
- softgre – конфигурация SoftGRE-туннелей<sup>1</sup>;
- vti – конфигурация VTI-туннелей;
- pptp – конфигурация PPTP-клиентов;
- pppoe – конфигурация PPPoE-клиентов;
- l2tp – конфигурация L2TP-клиентов;
- openvpn – конфигурация OpenVPN-клиентов.
- <NUM> – номер туннеля данного типа в конфигурации маршрутизатора.
- vlans – конфигурация VLAN;
- voice – конфигурация голосовых сервисов;
- vrf – конфигурация VRF;
- vrrp – конфигурация VRRP-протокола;
- without-wlc-ap – вывод всей конфигурации без отображения индивидуальных профилей ТД (ар <hh:hh:hh:hh:hh:hh>);
- wlc – конфигурация WLC:
  - without-wlc-ap – конфигурация WLC без отображения индивидуальных профилей ТД (ар <hh:hh:hh:hh:hh:hh>).
- wireless-controller – конфигурация параметров Wi-Fi контроллера;
- zabbix – конфигурация Zabbix-агента.

 <sup>1</sup> В текущей версии ПО данные функции поддерживаются только на маршрутизаторах ESR-1000.

## Необходимый уровень привилегий

10

## Командный режим

ROOT

## Пример:

```
esr# show candidate-config
ntp enable
ntp broadcast-client enable
syslog max-files 3
syslog file-size 512
syslog file default info
vlan 2
exit
security zone trusted
exit
security zone untrusted
exit
object-group service telnet
  port-range 23
exit
object-group service ssh
  port-range 22
exit
object-group service dhcp_server
  port-range 67
```



```
exit More? Enter - next line; Space - next page; Q - quit; R - show the rest.
```

## show configuration changes

Данной командой осуществляется просмотр различий между конфигурационными файлами.

### Синтаксис

```
show configuration changes [<CONFIG> < CONFIG>]
```

### Параметры

<CONFIG> – конфигурационный файл для сравнения. Могут принимать значения:

- candidate-config;
- running-config;
- factory-config;
- default-config;
- flash:backup/FILE.

### Значение по умолчанию

Без указания <CONFIG> отображается разница между running-config и candidate-config.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример:

```
esr(config-)# show configuration changes
+ interface gigabitethernet 1/0/1.100
+ ip firewall disable
+ ip address 10.54.22.1/24
+ exit
```

## show crypto certificates

Данная команда выводит информацию о количестве сертификатов.

### Синтаксис

```
show crypto certificates [ <CERTIFICATE-TYPE> [ <FILE> ] ]
```

### Параметры

<CERTIFICATE-TYPE> – тип сертификата или ключа, может принимать следующие значения:

- cert – X.509 сертификаты;
- dh – ключ Диффи-Хеллмана;
- private-key – приватные ключи;

- public-key – публичные ключи;
- ta – HMAC-ключ;
- crl – отозванные сертификаты;
- csr – запросы на сертификацию X.509;
- pfx – PKSC12-контейнер.

### Необходимый уровень привилегий

15

### Командный режим

ROOT

### Пример:

```
esr# show crypto certificates
Type          Total
-----
cert          3
dh            1
private-key   2
public-key    1
ta            1
crl           1
```

### show licence

Данная команда служит для просмотра информации об активной лицензии устройства.

### Синтаксис

show licence

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

5

### Командный режим

ROOT

### Пример:

```
wlc# show licence
Feature          Source      State      Value
Valid from      Expiries
-----
```

BRAS		File	Active	<b>true</b>
--	--			
BRAS		File	Candidate	<b>true</b>
--	--			
WLC		File	Active	<b>true</b>
--	--			
WLC		File	Candidate	<b>true</b>
--	--			
WLC-AP		File	Active	<b>100</b>
--	--			
WLC-AP		File	Candidate	<b>100</b>
--	--			

## show running-config

Данная команда служит для просмотра текущей конфигурации устройства.

### Синтаксис

```
show running-config [<SECTION>] [ full ]
```

### Параметры

<SECTION> – раздел конфигурации, описание приведено в разделе [show candidate-config](#).

full – ключ для отображения в конфигурации значений всех параметров, в том числе и ненастроенных.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример:

```
esr# show running-config syslog
syslog max-files 3
syslog file-size 512
syslog file default info
syslog console info
```

## show storage-devices

Данной командой выводится информация о подключенных внешних носителях (USB/MMC-карты памяти).

### Синтаксис

```
show storage-devices { usb | mmc | hdd [ smart ] }
```

### Параметры

usb – USB-накопитель информации;

mmc – SD/MMC-карта памяти;

hdd – HDD-жесткий диск;

smart – вывод технического состояния жесткого диска.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Примеры:

```
esr# show storage-devices mmc
```

Name	Total, MB	Used, MB	Free, MB
EF28-D074	99.79	72.64	27.15

```
wlc# show storage-devices hdd smart
```

```

HDD 1
~~~~~
ID      Name
Type    Status  Status  Decrypted  Value  Worst  Threshold  Raw  Flags
-----
1      raw-read-error-rate
Prefail Good    Good    --          200    200    51          0    47
3      spin-up-time
Prefail Good    Good    1.4 s      152    146    21          1400  39
4      start-stop-count
Old-age Good    Good    --          86     86     0           14732  50
5      reallocated-sector-count
Prefail Good    Good    0 sectors  200    200    140         0     51
7      seek-error-rate
Old-age Good    Good    --          200    200    0           0     46
9      power-on-hours
Old-age Good    Good    3.3 years  61     61     0           28886  50
10     spin-retry-count
Old-age Good    Good    --          100    100    0           0     50
11     calibration-retry-count
Old-age Good    Good    --          100    100    0           0     50
12     power-cycle-count
Old-age Good    Good    --          100    100    0           198   50
191    g-sense-error-rate
Old-age Good    Good    --          94     94     0           6     50
192    power-off-retract-count
Old-age Good    Good    --          200    200    0           163   50
193    load-cycle-count
Old-age Good    Good    --          192    192    0           26873  50
194    temperature-celsius-2
Old-age Good    Good    34.0 C    109    102    0           34    34

```

```

196  reallocated-event-count          200    200    0          0          50
Old-age  Good    Good    --
197  current-pending-sector            200    200    0          0          50
Old-age  Good    Good    0 sectors
198  offline-uncorrectable             100    253    0          0          48
Old-age  Good    Good    0 sectors
199  udma-crc-error-count              200    200    0          0          50
Old-age  Good    Good    --
200  multi-zone-error-rate             100    253    0          0          8
Bad sectors:    0
Overall status: good

```

## show usb

Данной командой выводится информация о подключенных USB-носителях.

### Синтаксис

```
show usb
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример:

```

esr-1000# show usb
USB device  Vendor      Product ID      Product version
-----
1-1         Flash       USB Disk       6.80

```

## show version

Данная команда служит для просмотра текущей версии программного обеспечения и аппаратной части устройства.

### Синтаксис

```
show version [ bl | hw | sw | voip ] [ unit { <RANGE-ID> | local } ]
```

### Параметры

<bl> – этот параметр позволяет просматривать только версию boot. Недоступно на vESR;

<hw> – этот параметр позволяет просматривать только версию аппаратной части устройства. Недоступно на vESR;

<sw> – этот параметр позволяет просматривать только версию программного обеспечения;

<voip> – этот параметр позволяет просматривать только версию VoIP. Доступно только на устройствах ESR-12V, ESR-12VF и ESR-15VF;

<RANGE-ID> – принимает номера юнитов в формате: ID,ID-ID. Номер юнита принимает значения [1..2];

local – ключ для отображения версии устройства, на котором вводится команда, в формате вывода без включенного кластера.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример:

```
esr# show version
Boot version:
  1.24.0.1 (2024-08-01 13:40:59)
SW version:
  1.24.0 build 1[d9bdbda] (2024-08-01 18:06:41)
HW version:
  1v7

esr# show version bl
Boot version:
  1.24.0.1 (2024-08-01 13:40:59)

esr# show version hw
HW version:
  1v7

esr# show version sw
SW version:
  1.24.0 build 1[d9bdbda] (2024-08-01 18:06:41)
```

## time-period

Данной командой задаётся период времени, по истечении которого будет осуществляться автоматическое резервирование конфигурации.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

time-period <TIME>

no time-period

## Параметры

<TIME> – периодичность автоматического резервирования конфигурации, принимает значение в минутах [1..525600].

**Значение по умолчанию**

720 минут.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-ARCHIVE

**Пример:**

```
esr(config-archive)# time-period 1440
```

**type**

Данной командой устанавливается тип сохранения резервных конфигураций маршрутизатора. При использовании отрицательной формы команды (no) устанавливается режим по умолчанию.

**Синтаксис**

```
type <TYPE>
```

```
no type
```

**Параметры**

<TYPE> – тип сохранения резервных конфигураций маршрутизатора. Принимает значения:

- local – сохранения резервных конфигураций происходит в раздел flash:backup/ с именем файла вида "config\_YYYYMMDD\_HHMMSS\_username;" где:
  - YYYY – год, согласно системным часам маршрутизатора на момент записи резервной копии конфигурации;
  - MM – месяц, согласно системным часам маршрутизатора на момент записи резервной копии конфигурации;
  - DD – день, согласно системным часам маршрутизатора на момент записи резервной копии конфигурации;
  - HH – час, согласно системным часам маршрутизатора на момент записи резервной копии конфигурации;
  - MM – минута, согласно системным часам маршрутизатора на момент записи резервной копии конфигурации;
  - SS – секунда, согласно системным часам маршрутизатора на момент записи резервной копии конфигурации.
  - username – имя учетной записи, в рамках сессии которой был выполнен «commit».
- remote – сохранения резервных конфигураций происходят на удаленный сервер;
- both – сохранения резервных конфигураций в раздел flash:backup и на удаленный сервер.

**Значение по умолчанию**

remote

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-ARCHIVE

**Пример:**

```
esr(config-archive)# type both
```

**unmount storage-device**

Данной командой производится отключение внешнего накопителя.

**Синтаксис**

```
unmount storage-device { usb | mmc | hdd }
```

**Параметры**

usb – USB-накопитель информации.

mmc – SD-/MMC-карта памяти.

hdd – HDD-жесткий диск.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример:**

```
esr# show storage-devices usb
esr# 2024-04-27T08:55:40+00:00 %USBFLASH-I-CHANGE: 'D2E5-006B' has been inserted!
```

**update crypto default**

Данной командой обновляются сертификаты.

**Синтаксис**

```
update crypto default <CERTIFICATE-TYPE> [invalid-after] [valid-after]
```

**Параметры**

<CERTIFICATE-TYPE> – тип сертификата или ключа, может принимать следующие значения:



- dh – ключ Диффи-Хеллмана;
- ca – корневой сертификат;
- cert – X.509-сертификат.

invalid-after – установка даты, до конца которой будет действовать сертификат;

valid-after – установка даты, от начала которой будет действовать сертификат.

### Необходимый уровень привилегий

15

### Командный режим

ROOT

### Пример:

```
esr# update crypto default ca invalid-after 00:00:00 01 january 2025
esr# show crypto certificates cert default_ca.pem | i after
Valid after:                2024-06-13 12:27:57
Invalid after:              2025-01-01 00:00:00
```

## 7 Настройка TFTP-сервера

- [dscp](#)
- [enable](#)
- [ip tftp server](#)
- [local-port](#)

### **dscp**

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов TFTP-сервера.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

#### **Синтаксис**

```
dscp <DSCP>  
no dscp
```

#### **Параметры**

<DSCP> – значение кода DSCP, принимает значение в диапазоне [0..63].

#### **Значение по умолчанию**

0

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG-TFTP-SERVER

#### **Пример:**

```
esr(config-tftp-server)# dscp 22
```

### **enable**

Данной командой включается TFTP-сервер на маршрутизаторе.

Использование отрицательной формы команды (no) отключает TFTP-сервер.

#### **Синтаксис**

```
[no] enable
```

#### **Параметры**

Отсутствуют.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-TFTP-SERVER

**Пример:**

```
esr(config-tftp-server)# enable
```

**ip tftp server**

Данная команда переводит маршрутизатор в режим конфигурирования TFTP-сервера.

Рабочим разделом TFTP-сервера является flash:data/.

Использование отрицательной формы команды (no) выключает и убирает настройки TFTP-сервера.

**Синтаксис**

```
ip tftp server vrf <VRF>
```

**Параметры**

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать TFTP-сервер.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример:**

```
esr(config)# ip tftp server
```

**local-port**

Данной командой определяется порт TFTP-сервера на маршрутизаторе.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
local-port <PORT>
```

**Параметры**

<PORT> – номер порта, принимает значение в диапазоне [1..65535].

**Значение по умолчанию**

69

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-TFTP-SERVER

**Пример:**

```
esr(config-tftp-server)# local-port 6212
```

## 8 Настройка DNS

- [domain ip host](#)
- [domain lookup enable](#)
- [domain name](#)
- [domain name-server](#)

### domain ip host

Данная команда определяет статическую DNS-запись.

Использование отрицательной формы команды (no) удаляет запись.

#### Синтаксис

```
[no] domain ip host <NAME> <IP>
```

#### Параметры

<NAME> – имя хоста, задаётся строкой до 253 символов.

<IP> – IP-адрес хоста, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG

#### Пример:

```
esr(config)# domain ip host eltex.loc 172.16.0.3
```

### domain lookup enable

Данная команда включает разрешение DNS-имен.

Использование отрицательной формы команды (no) выключает разрешение DNS-имен.

#### Синтаксис

```
[no] domain lookup enable [ vrf <VRF> ]
```

#### Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет включено разрешение DNS-имен.

#### Необходимый уровень привилегий

10

## Командный режим

CONFIG

### Пример:

```
esr(config)# domain lookup enable
```

## domain name

Команда позволяет назначить имя домена для маршрутизатора.

Использование отрицательной формы команды (no) удаляет имя домена для маршрутизатора.

### Синтаксис

```
domain name <NAME>
```

```
no domain name
```

### Параметры

<NAME> – имя домена маршрутизатора, задаётся строкой от 1 до 31 символа.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG

### Пример:

```
esr(config)# domain name eltex-co.ru
```

## domain name-server

Данная команда определяет IP-адрес DNS-сервера, используемого для разрешения DNS-имен.

Использование отрицательной формы команды (no) удаляет адрес DNS-сервера.

### Синтаксис

```
[no] domain name-server <IP> [vrf <VRF>]
```

### Параметры

<IP> – IP-адрес используемого DNS-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет происходить взаимодействие с DNS-сервером.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример:**

```
esr(config)# domain name-server 172.16.0.1
```

## 9 Настройка WLC

- enable
- wlc
- outside-address
- Настройки сервиса Airtune
  - airtune
    - enable
    - port
    - report delete-time
  - airtune-profile
    - 802.11k disable
    - 802.11r disable
    - 802.11r over-ds disable
    - 802.11r reassoc-deadline
    - 802.11v disable
    - 802.11v load-balance-interval
    - away-timeout
    - blacklist threshold high
    - blacklist threshold low
    - blacklist timeout
    - client max-age
    - client rssi-renew
    - client try-no-balance
    - dca disable
    - dca threshold
    - eltex-rrm-scan disable
    - hd-mode
    - hysteresis
    - load-balance balance-enterprise disable
    - load-balance disable
    - load-balance sta high
    - load-balance sta low
    - optimization candidate
    - optimization mode
    - optimization time
    - report days-to-live
    - report disable
    - tpc disable
    - tpc rssi threshold
- Настройка индивидуального профиля точки доступа
  - ap
    - airtune-disable
    - ap-location
    - ap-model
    - description
    - hostname
    - override ap-profile
    - override 2g
    - override 5g
      - bandwidth
      - channel
      - limit-channels
      - rates-supported
      - rates-basic
      - tx-power
      - work-mode



- override wids
  - shared-key
  - wids-profile
  - white-list
  - black-list
- wids-disable
- Настройка профиля локации
  - ap-location
    - airtune-profile
    - ap-profile
    - description
    - mode tunnel
    - radio-2g-profile
    - radio-5g-profile
    - ssid-profile
    - timezone
    - wids
      - shared-key
      - wids-profile
      - white-list
      - black-list
      - wids-disable
- Настройка профиля общих настроек точек доступа
  - ap-profile
    - captive-portal
      - ap-ip-alias
      - web-redirector
    - description
    - password
    - services
      - ip http port
      - ip http service
      - ip https port
      - ip ssh port
      - ip ssh service
      - ip telnet port
      - ip telnet service
      - snmp-server
      - snmp-server community
      - snmp-server host
      - z-wave
      - z-wave address
      - z-wave port
      - z-wave secure
- Настройка профилей параметров радио точек доступа
  - radio-2g-profile
  - radio-5g-profile
    - rates-supported
    - rates-basic
    - bandwidth
    - description
    - dfs
    - ip dhcp information option action
    - ip dhcp information option format-type circuit-id
    - ip dhcp information option format-type remote-id
    - ip dhcp information option format-type mac-address

- limit-channels
- obss-coexistence
- tx-power
- work-mode
- Настройка адресного пространства
  - ip-pool
    - ap-location
    - description
    - network
- Настройка фильтрации логов
  - log-filter
- Настройка использования портала
  - portal-profile
    - age-timeout
    - description
    - redirect-url
    - redirect-url-custom
    - verification-mode
    - virtual-portal-name
    - white-list
- Настройка профиля для RADIUS-сервера
  - radius-profile
    - acct-address
    - acct-enable
    - acct-interval
    - acct-password
    - acct-periodic
    - acct-port
    - auth-acct-id-send
    - auth-address
    - auth-password
    - auth-port
    - description
    - domain
    - nas-id
    - tls-enable
- Настройка профиля ограничения скорости
  - policy-profile ap
    - description
    - rate-limit
- Настройка сервис-активатора
  - service-activator
    - aps join auto
    - password private-crt-key
    - port
- Настройка SSID
  - ssid-profile
    - 802.11kv
    - 802.11r
    - band
    - band-steer-mode
    - check-signal-enable
    - check-signal-timeout
    - description
    - enable
    - general-vlan-id

- general-vlan-mode
- hidden
- inactivity-timeout
- key-wpa
- local-switching
- mac-auth mode local
- mac-auth mode radius
- minimal-signal
- pmksa-caching
- policy-profile ap
- portal-enabled
- portal-profile
- priority-by-dscp
- radius-profile
- roaming-signal
- security-mode
- ssid
- sta-limit
- station-isolation
- vlan-id
- vlan-priority
- vlan-trunk
- Настройки сервиса WIDS/WIPS
  - wids
    - enable
    - shared-key
    - wids-profile
    - white-list
    - black-list
  - wids-profile
    - bruteforce-detection
      - enable
      - threshold
      - interval
      - mac-ban enable
      - mac-ban timeout
    - description
    - dos-detection
      - enable
      - interval
      - trap-send-period
      - threshold assoc
      - threshold auth
      - threshold beacon
      - threshold blockack
      - threshold blockack-req
      - threshold cts
      - threshold deauth
      - threshold disassoc
      - threshold leap
      - threshold probe
      - threshold ps-poll
      - threshold reassoc
      - threshold rts
    - scan
      - interface

- mode
  - passive interval
  - passive time
  - sentry time
- prevention-mode
- attack-stats-trap-send-period
- Настройка менеджера обновления ПО
  - update-manager
    - allow-update-with-clients
    - end-time
    - scheduled
    - start-time
- Настройка резервирования
  - failover
- Команды просмотра
  - show wlc
  - show wlc airtune
  - show wlc airtune roaming statistics
  - show wlc airtune rrm optimization report
  - show wlc airtune rrm statistics
  - show wlc airtune sessions
  - show wlc configuration warnings
  - show wlc ap
  - show wlc ap interfaces
  - show wlc ap radios
  - show wlc ap vap
  - show wlc ap firmware
  - show wlc clients
  - show wlc journal ap
  - show wlc journal clients
  - show wlc journal wids
  - show wlc service-activator aps
- Команды управления
  - clear wlc airtune session
  - clear wlc ap
  - clear wlc client
  - clear wlc journal
  - join wlc ap
  - reload wlc airtune rrm optimization
  - reload wlc ap
  - set wlc indication enable ap
  - set wlc indication disable ap
- Настройка логирования на точках доступа
  - autolog address
  - autolog enable
  - autolog logfile-limit
  - autolog period
  - autolog retry
  - syslog disable
  - syslog mode
  - syslog port
  - syslog server
  - syslog size
  - trace
    - acsd debug-5g
    - acsd debug-chanim

- `acsd debug-dfsr`
- `acsd enable`
- `acsd logfile`
- `acsd logfile-limit`
- `acsd loglevel`
- `no acsd all`
- `airtune enable`
- `airtune logfile`
- `airtune logfile-limit`
- `airtune loglevel`
- `no airtune all`
- `bandsteer enable`
- `bandsteer logfile`
- `bandsteer logfile-limit`
- `bandsteer loglevel`
- `bandsteer sta-mac`
- `no bandsteer all`
- `captive-portal enable`
- `captive-portal logfile`
- `captive-portal logfile-limit`
- `captive-portal loglevel`
- `no captive-portal all`
- `captive-portal apbd enable`
- `captive-portal apbd logfile`
- `captive-portal apbd logfile-limit`
- `captive-portal apbd loglevel`
- `no captive-portal apbd all`
- `captive-portal redirector-debug`
- `captive-portal tinyproxy enable`
- `captive-portal tinyproxy logfile`
- `captive-portal tinyproxy logfile-limit`
- `no captive-portal tinyproxy all`
- `configd enable`
- `configd logfile`
- `configd logfile-limit`
- `no configd all`
- `dmesg enable`
- `dmesg logfile`
- `dmesg logfile-limit`
- `no dmesg all`
- `ftd enable`
- `ftd logfile`
- `ftd logfile-limit`
- `no ftd all`
- `hostapd enable`
- `hostapd logfile`
- `hostapd logfile-limit`
- `hostapd loglevel`
- `no hostapd all`
- `monitord enable`
- `monitord logfile`
- `monitord logfile-limit`
- `monitord loglevel`
- `no monitord all`
- `netconf enable`
- `netconf logfile`

- netconf logfile-limit
- no netconf all
- networkd enable
- networkd logfile
- networkd logfile-limit
- networkd loglevel
- no networkd all
- snmp enable
- snmp logfile
- snmp logfile-limit
- no snmp all
- wlc-activator enable
- wlc-activator logfile
- wlc-activator logfile-limit
- no wlc-activator all
- wlc-activator server enable
- wlc-activator server logfile
- wlc-activator server logfile-limit
- no wlc-activator server all
- wids enable
- wids logfile
- wids logfile-limit
- wids loglevel
- no wids all
- Настройка журналирования событий WLC
  - wlc-journal
  - limit

## enable

Данная команда предназначена для активации функций и сервисов контроллера или точек доступа. Использование отрицательной формы команды (no) отключает функции.

## Синтаксис

```
[no] enable
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Отключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC

CONFIG-WLC-SSID-PROFILE

CONFIG-WLC-AIRTUNE

## CONFIG-WLC-WIDS

### Пример

```
wlc(config-wlc)# enable
```

### wlc

Данная команда осуществляет переход к конфигурированию контроллера.

Использование отрицательной формы команды (no) удаляет сконфигурированные настройки.

### Синтаксис

```
[no] wlc
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
wlc(config)# wlc  
wlc(config-wlc)#
```

### outside-address

Данная команда устанавливает IP-адрес контроллера, который будет доступен точкам доступа. Точки доступа будут подключаться к этому адресу для работы с сервис-активатором и Airtune.

Использование отрицательной формы команды (no) удаляет заданный IP-адрес.

### Синтаксис

```
outside-address <ADDR>  
no outside-address
```

### Параметры

<ADDR> – IP-адрес контроллера, задается в виде: AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC

**Пример**

```
wlc(config-wlc)# outside-address 192.168.1.1
```

**Настройки сервиса Airtune****airtune**

Данная команда позволяет перейти в общие настройки сервиса Airtune.

Использование отрицательной формы команды (no) устанавливает значение общих настроек сервиса по умолчанию.

**Синтаксис**

[no] airtune

**Параметры**

Отсутствует.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC

**Пример**

```
wlc(config-wlc)# airtune  
wlc(config-wlc-airtune)#
```



**enable**

Данная команда предназначена для активации функций и сервисов контроллера или точек доступа. Использование отрицательной формы команды (no) отключает функции.

**Синтаксис**

```
[no] enable
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC

CONFIG-WLC-SSID-PROFILE

CONFIG-WLC-AIRTUNE

CONFIG-WLC-WIDS

**Пример**

```
wlc(config-wlc-wids)# enable
```

**port**

Данная команда указывает номер порта, на котором работает сервис.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
port <NUMBER>
```

```
no port
```

**Параметры**

<NUMBER> – номер порта, принимает значения от 1024 до 65535.

**Значение по умолчанию**

8099

**Необходимый уровень привилегий**

10

## Командный режим

CONFIG-WLC-AIRTUNE

### Пример

```
wlc(config-wlc-airtune)# port 8099
```

### report delete-time

Данная команда задает время удаления отчета.

Использование отрицательной формы команды (no) сбрасывает параметр на значение по умолчанию.

### Синтаксис

```
report delete-time <HH:MM>  
no report
```

### Параметры

<HH:MM> – время удаления отчета, где первые две цифры указывают час, а вторые – минуты.

### Значение по умолчанию

21:00

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AIRTUNE

### Пример

```
wlc(config-wlc-airtune)# report delete-time 21:00
```

### airtune-profile

Данная команда позволяет перейти в настройки профиля Airtune.

Использование отрицательной формы команды (no) удаляет созданный профиль.

### Синтаксис

```
airtune-profile <NAME>  
no airtune { <NAME> | all }
```

### Параметры

<NAME> – название профиля, задается строкой до 235 символов;

all – параметр для удаления всех профилей.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC

**Пример**

```
wlc(config-wlc)# airtune-profile test
wlc(config-airtune-profile)#
```

**802.11k disable**

Данная команда отключает синхронизацию списков для роуминга стандарта 802.11k.

Использование отрицательной формы команды (no) включает синхронизацию списков для роуминга стандарта 802.11k.

Роуминг по протоколу 802.11k может быть организован между любыми сетями (открытые/шифрованные). Если на точке доступа настроена работа по протоколу 802.11k, то при подключении клиента точка доступа передает ему список «дружественных» точек доступа, на которые клиент может переключиться в процессе роуминга. Список содержит информацию о MAC-адресах точек доступа и каналов, на которых они работают.

Использование 802.11k позволяет сократить время, которое клиент затрачивает на поиск другой сети при роуминге, так как клиенту не нужно производить сканирование каналов, на которых нет целевых точек доступа доступных для переключения.

Данный вид роуминга доступен только для тех клиентских устройств, которые поддерживают 802.11k.

**Синтаксис**

[no] 802.11k disable

**Параметры**

Отсутствуют.

**Значение по умолчанию**

Включено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-AIRTUNE-PROFILE

## Пример

```
wlc(config-airtune-profile)# 802.11k disable
```

### 802.11r disable

Данная команда отключает отправку ключей для роуминга стандарта 802.11r.

Использование отрицательной формы команды (no) включает отправку ключей для роуминга стандарта 802.11r.

Данный вид роуминга доступен только для тех клиентских устройств, которые поддерживают 802.11r.

Роуминг 802.11r возможен только между VAP с режимом безопасности WPA2-Personal и WPA2-Enterprise.

### Синтаксис

```
[no] 802.11r disable
```

### Параметры

Отсутствуют.

### Значение по умолчанию

Включено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-AIRTUNE-PROFILE

## Пример

```
wlc(config-airtune-profile)# 802.11r disable
```

### 802.11r over-ds disable

Данная команда выключает функционал Over-DS стандарта 802.11r. Если параметр отключен, используется Over-Air.

Использование отрицательной формы команды (no) включает функционал Over-DS стандарта 802.11r.

### Синтаксис

```
[no] 802.11r over-ds disable
```

### Параметры

Отсутствуют.

**Значение по умолчанию**

Включено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-AIRTUNE-PROFILE

**Пример**

```
wlc(config-airtune-profile)# 802.11k over-ds disable
```

**802.11r reassoc-deadline**

Данная команда настраивает максимальный период времени, в течение которого ТД должны обмениваться данными о попытке роуминга клиента (RRB-пакеты). Если ответ на запрос по истечению таймаута не пришел, RRB-запрос на бесшовный роуминг считается неуспешным.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
802.11r reassoc-deadline <TIME>
no 802.11r reassoc-deadline
```

**Параметры**

<TIME> – временной промежуток, доступны значения от 1000 до 268431360 миллисекунд.

**Значение по умолчанию**

1000

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-AIRTUNE-PROFILE

**Пример**

```
wlc(config-airtune-profile)# 802.11k reassoc-deadline 1000
```

**802.11v disable**

Данная команда отключает функционал работы роуминга стандарта 802.11v. Функционал работает совместно со стандартом 802.11k, роуминг осуществляется по общим спискам. Для его работы в SSID-профиле должна быть включена опция 802.11k/v.

Помимо помощи роумингу клиента в соответствии со стандартом 802.11v, активируется работа

балансировки клиентов.

Переключение клиентов на другие ТД работает в зависимости от пороговых значений, установленных в параметрах *"load-balance sta high"* и *"load-balance sta low"*.

Если клиентов на ТД стало более чем порог *"load-balance sta low"*, то AirTune понизит приоритет данной ТД в списках 802.11k/v для подключения новых пользователей, и разошлет информацию об этом всем её соседям.

Если клиентов на ТД стало более чем порог *"load-balance sta high"*, то AirTune понизит приоритет данной ТД до нуля и исключит ее из рассылки возможных ТД для роуминга. А также начнет рекомендовать "нагруженной" ТД переключить нескольких клиентов на соседние ТД по стандарту 802.11v (отправкой сообщения BSS Transition Request).

Использование отрицательной формы команды (no) включает настройку 802.11v.

### Синтаксис

```
[no] 802.11v disable
```

### Параметры

Отсутствуют.

### Значение по умолчанию

Включено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-AIRTUNE-PROFILE

### Пример

```
wlc(config-airtune-profile)# 802.11v disable
```

### 802.11v load-balance-interval

Данная команда задает период балансировки клиентов между точками доступа в рамках стандарта 802.11v.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
802.11v load-balance-interval <TIME>
```

```
no 802.11v load-balance-interval
```

### Параметры

<TIME> – временной промежуток, допустимы значения от 0 до 86400 секунд.

### Значение по умолчанию

60

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-AIRTUNE-PROFILE

## Пример

```
wlc(config-airtune-profile)# 802.11v load-balance-interval 100
```

## away-timeout

Данная команда устанавливает защитный интервал для определения доступности ТД сервером, то есть допустимое время ожидания ТД в случае потери связи, по истечении которого сервис будет считать ТД отключенной от сервиса.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
away-timeout <TIME>  
no away-timeout
```

## Параметры

<TIME> – временной интервал, допустимы значения от 10 до 3600 секунд.

## Значение по умолчанию

120

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AIRTUNE-PROFILE

## Пример

```
wlc(config-airtune-profile)# away-timeout 120
```

## blacklist threshold high

Данная команда устанавливает верхнюю границу окончания зоны устойчивого приема сигнала от клиента, то есть порог уровня RSSI от клиента, при превышении которого подключенный клиент будет считаться в "уверенной" зоне и поиск новой ТД не начнется в случае, если ТД не перегружена.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
blacklist threshold high <RSSI>  
no blacklist threshold high
```

### Параметры

<RSSI> – порог уровня RSSI, допустимы значения от -100 до 1 Дбм.

### Значение по умолчанию

-65

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-AIRTUNE-PROFILE

### Пример

```
wlc(config-airtune-profile)# blacklist threshold high -65
```

### blacklist threshold low

Данная команда устанавливает нижнюю границу окончания зоны устойчивого приема сигнала от клиента, то есть порог уровня RSSI от клиента, в случае если уровень от клиента меньше указанного в данном параметре, клиент считается находящимся в "неуверенной" зоне. Сервис будет пытаться найти для клиента ТД с "уверенным" приемом, и последующим переключением клиента на целевую ТД.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
blacklist threshold low <RSSI>  
no blacklist threshold low
```

### Параметры

<RSSI> – порог уровня RSSI, допустимы значения от -100 до 1 Дбм.

### Значение по умолчанию

-75

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-AIRTUNE-PROFILE



## Пример

```
wlc(config-airtune-profile)# blacklist threshold low -75
```

## blacklist timeout

Данная команда устанавливает время блокировки клиента, в течение которого работает балансировка. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
blacklist timeout <TIME>  
no blacklist timeout
```

## Параметры

<TIME> – временной интервал, допустимы значения от 5 до 3600 секунд.

## Значение по умолчанию

30

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AIRTUNE-PROFILE

## Пример

```
wlc(config-airtune-profile)# blacklist timeout 30
```

## client max-age

Данная команда устанавливает максимально допустимое время прошедшее с момента последнего prob-запроса от клиента.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
client max-age <TIME>  
no client max-age
```

## Параметры

<TIME> – временной интервал, допустимы значения от 0 до 300 секунд.

## Значение по умолчанию

20

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AIRTUNE-PROFILE

**Пример**

```
wlc(config-airtune-profile)# client max-age 20
```

**client rssi-renew**

Данная команда устанавливает защитный интервал для определения зоны приема сигнала от клиента. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
client rssi-renew <TIME>  
no client rssi-renew
```

**Параметры**

<TIME> – временной интервал, допустимы значения от 1 до 300 секунд.

**Значение по умолчанию**

15

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AIRTUNE-PROFILE

**Пример**

```
wlc(config-airtune-profile)# client rssi-renew 15
```

**client try-no-balance**

Данная команда устанавливает количество игнорируемых попыток подключения, заблокированного клиента.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
client try-no-balance <COUNT>  
no client try-no-balance
```

## Параметры

<COUNT> – количество попыток, допустимы значения от 1 до 10.

## Значение по умолчанию

2

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AIRTUNE-PROFILE

## Пример

```
wlc(config-airtune-profile)# client try-no-balance 2
```

## dca disable

Данная команда выключает алгоритм автоматического распределения частотных каналов точек доступа.

Использование отрицательной формы команды (no) активирует функционал динамического распределения для точек доступа, чтобы избежать интерференции.

## Синтаксис

[no] dca disable

## Параметры

Отсутствуют.

## Значение по умолчанию

Включено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-AIRTUNE-PROFILE

## Пример

```
wlc(config-airtune-profile)# dca disable
```

## dca threshold

Данная команда устанавливает порог изменения каналов при динамическом распределении каналов.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
dca threshold <VALUE>
no dca threshold
```

### Параметры

<VALUE> – порог изменения, допустимые значения от 0 до 99%.

### Значение по умолчанию

25

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-AIRTUNE-PROFILE

### Пример

```
wlc(config-airtune-profile)# dca threshold 25
```

### eltex-rrm-scan disable

Данная команда выключает ускоренное сканирование.

Использование отрицательной формы команды (no) включает ускоренное сканирование.

С включенным параметром ТД в один момент времени обмениваются специальными Action-фреймами в определенном частотном канале, который сообщил им сервис. По окончании обмена передают сообщение на сервис с полученными результатами. Весь процесс оптимизации в таком режиме будет занимать не более пары минут вне зависимости от количества ТД в домене.

В случае отключенного параметра ТД по очереди сканируют все каналы, учитывают влияние конкурентных ТД. В данном случае время, требуемое для оптимизации, будет увеличиваться при увеличении количества ТД (на 1 ТД – 50-60 секунд).

### Синтаксис

```
[no] eltex-rrm-scan disable
```

### Параметры

Отсутствуют.

### Значение по умолчанию

Включено.

### Необходимый уровень привилегий

10

**Командный режим**

CONFIG-AIRTUNE-PROFILE

**Пример**

```
wlc(config-airtune-profile)# eltex-rrm-scan disable
```

**hd-mode**

Данная команда включает режим DCA-HD. Мощность ТД управляется только на ТД, работающих на одинаковых каналах.

Использование отрицательной формы команды (no) выключает режим HD.

**Синтаксис**

```
[no] hd-mode
```

**Параметры**

Отсутствуют.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-AIRTUNE-PROFILE

**Пример**

```
wlc(config-airtune-profile)# hd-mode
```

**hysteresis**

Данная команда устанавливает допустимую погрешность для частотного диапазона 2,4 ГГц или 5 ГГц. Разрешенная "погрешность" от порога, если полученный сигнал лежит в диапазоне RSSI Treshold+-2dbm, оптимизацию по мощности проводить не следует.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
hysteresis { 2g <VALUE> | 5g <VALUE> }
no hysteresis { 2g | 5g }
```

**Параметры**

<VALUE> – допустимая погрешность, допустимы значения от 1 до 10.

**Значение по умолчанию**

2

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-AIRTUNE-PROFILE

**Пример**

```
wlc(config-airtune-profile)# hysteresis 2g 2
```

**load-balance balance-enterprise disable**

Данная команда выключает балансировку клиентов enterprise-сетей между ТД.

Использование отрицательной формы команды (no) включает балансировку.

Функционал нужен для равномерного распределения клиентов между ТД, чтобы избежать перегрузки одной из ТД, если в зоне видимости клиента есть более свободная ТД.

**Синтаксис**

```
[no] load-balance balance-enterprise disable
```

**Параметры**

Отсутствуют.

**Значение по умолчанию**

Включено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-AIRTUNE-PROFILE

**Пример**

```
wlc(config-airtune-profile)# load-balance balance-enterprise disable
```

**load-balance disable**

Данная команда выключает балансировку по всем ТД в домене, независимо от их фактического расположения.

Использование отрицательной формы команды (no) включает балансировку.

Функционал нужен для равномерного распределения клиентов между ТД, чтобы избежать перегрузки одной из ТД, если в зоне видимости клиента есть более свободная ТД.

### Синтаксис

```
[no] load-balance disable
```

### Параметры

Отсутствуют.

### Значение по умолчанию

Включено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-AIRTUNE-PROFILE

### Пример

```
wlc(config-airtune-profile)# load-balance disable
```

### load-balance sta high

Данная команда устанавливает порог количества подключенных клиентов на радиоинтерфейсе, при превышении которого точка будет считаться перегруженной.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
load-balance sta high <COUNT>  
no load-balance sta high
```

### Параметры

<COUNT> – количество клиентов, допустимы значения от 1 до 100.

### Значение по умолчанию

20

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-AIRTUNE-PROFILE

## Пример

```
wlc(config-airtune-profile)# load-balance sta high 20
```

### load-balance sta low

Данная команда устанавливает порог количества подключенных клиентов на радиointерфейсе, при превышении которого сервис будет искать для новых клиентов более свободную ТД (если таковая не найдется, клиент продолжит работу на текущей точке доступа). Если количество клиентов меньше текущего порога – точка доступа считается свободной.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
load-balance sta low <COUNT>
no load-balance sta low
```

### Параметры

<COUNT> – количество клиентов, допустимы значения от 1 до 100.

### Значение по умолчанию

5

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-AIRTUNE-PROFILE

## Пример

```
wlc(config-airtune-profile)# load-balance sta low 5
```

### optimization candidate

Данная команда позволяет выбрать режим балансировки и роуминга между всеми ТД, независимо от их фактического расположения или только между соседствующими ТД.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
optimization candidate [ neighbors | all ]
no optimization candidate
```

### Параметры

neighbors – ТД будут сканировать эфир и определять какие ТД являются соседями, чтобы балансировать клиентов и осуществлять роуминг только между рядом стоящими ТД (меньше лишнего трафика в проводной сети, но больше в радио среде);



all – сервис использует функционал в рамках всего домена, даже если ТД находятся на большом расстоянии друг от друга (больше трафика в проводной сети, меньше в радио среде).

### Значение по умолчанию

all

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-AIRTUNE-PROFILE

### Пример

```
wlc(config-airtune-profile)# optimization candidate all
```

### optimization mode

Данная команда позволяет выбрать параметр, по которому будет срабатывать оптимизация. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
optimization mode [ event | time | full | none ]  
no optimization mode
```

### Параметры

event – включение функционала оптимизации по событию:

- Добавление новой ТД в домен;
- Удаление ТД из домена;
- Пропадание связи до одной из ТД более 5 минут;
- Изменение конфигурации радио параметров на ТД в домене;

time – включение функционала оптимизации по указанному времени;

full – включение функционала оптимизации и по событию, и по указанному времени;

none – выключение функционала оптимизации.

### Значение по умолчанию

event

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-AIRTUNE-PROFILE

## Пример

```
wlc(config-airtune-profile)# optimization mode event
```

## optimization time

Данная команда позволяет указать время, по которому будет срабатывать оптимизация. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
optimization time <TIME>  
no optimization time
```

## Параметры

<TIME> – время срабатывания оптимизации, задается в формате ЧЧ:ММ, где первые две цифры – это часы, вторые – минуты.

## Значение по умолчанию

00:00

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-AIRTUNE-PROFILE

## Пример

```
wlc(config-airtune-profile)# optimization time 00:00
```

## report days-to-live

Данная команда указывает время жизни отчетов об оптимизации RRM. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
report days-to-live <DAYS>  
no report days-to-live
```

## Параметры

<DAYS> – время жизни отчетов, допустимы значения от 1 до 365 дней.

## Значение по умолчанию

93

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-AIRTUNE-PROFILE

**Пример**

```
wlc(config-airtune-profile)# report days-to-live 93
```

**report disable**

Данная команда выключает генерацию отчетов работы RRM.

Использование отрицательной формы команды (no) включает генерацию отчетов RRM.

**Синтаксис**

```
[no] report disable
```

**Параметры**

Отсутствуют.

**Значение по умолчанию**

Включено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-AIRTUNE-PROFILE

**Пример**

```
wlc(config-airtune-profile)# report disable
```

**tpc disable**

Данная команда выключает функцию автоматического управления мощностью.

Использование отрицательной формы команды (no) включает автоматическое управления мощностью.

**Синтаксис**

```
[no] tpc disable
```

**Параметры**

Отсутствуют.

**Значение по умолчанию**

Включено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-AIRTUNE-PROFILE

**Пример**

```
wlc(config-airtune-profile)# tpc disable
```

**tpc rssi threshold**

Данная команда регулирует максимальный уровень сигнала, с которым соседние ТД могут видеть друг друга в диапазоне 2.4 ГГц или 5 ГГц. В зависимости от полученного уровня RSSI от соседней ТД, сервер будет сравнивать его с RSSI Treshold и рекомендовать уменьшить ( $RSSI_{ТД} > RSSI\_Treshold$ ) либо увеличить мощность передатчика ( $RSSI_{ТД} < RSSI\_Treshold$ ).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
tpc rssi threshold { 2g <RSSI> | 5g <RSSI> }
```

**Параметры**

<RSSI> – максимальный уровень сигнала, Дбм, допустимые значения от -100 до -1.

**Значение по умолчанию**

2g: -70

5g: -65

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-AIRTUNE-PROFILE

**Пример**

```
wlc(config-airtune-profile)# tpc rssi threshold 2g -70
```

## Настройка индивидуального профиля точки доступа

### ap

Данная команда добавляет в систему точку доступа по MAC-адресу.

Использование отрицательной формы команды (no) удаляет точку доступа.

### Синтаксис

```
ap <MAC-ADDRESS>  
no ap { <MAC-ADDRESS> | all }
```

### Параметры

<MAC-ADDRESS> – mac-адрес точки доступа в формате HH:HH:HH:HH:HH:HH;

all – команда удаляет все добавленные точки доступа.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC

### Пример

```
wlc(config-wlc)# ap 00:00:00:00:00:00
```

### airtune-disable

Данная команда отключает работу сервиса Airtune для выбранной точки доступа.

Использование отрицательной формы команды (no) включает работу сервиса Airtune для выбранной точки доступа.

### Синтаксис

```
[no] airtune-disable
```

### Параметры

Отсутствует.

### Значение по умолчанию

Включено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP

**Пример**

```
wlc(config-wlc-ap)# airtune-disable
```

**ap-location**

Данная команда создает/назначает профиль локации, который описывает используемое оборудование и SSID.

Использование отрицательной формы команды (no) удаляет профиль.

**Синтаксис**

```
ap-location <NAME>  
no ap-location { <NAME> | all }
```

**Параметры**

<NAME> – название профиля локации, задается строкой до 235 символов;

all – команда удаляет все созданные профили.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC

CONFIG-WLC-AP

CONFIG-WLC-IP-POOL

**Пример**

```
wlc(config-wlc)# ap-location default
```

**ap-model**

Данная команда указывает модель точки доступа.

Использование отрицательной формы команды (no) удаляет установленное значение.

**Синтаксис**

```
ap-model <BOARD-TYPE>
no ap-model
```

**Параметры**

<BOARD-TYPE> – тип точки доступа, доступные значения:

- WEP-1L
- WEP-2L
- WEP-3L
- WEP-200L
- WEP-30L
- WEP-30L-Z
- WEP-3ax
- WOP-2L
- WOP-20L
- WOP-30L
- WOP-30LI
- WOP-30LS
- WEP-2ac
- WEP-2ac Smart
- WOP-2ac
- WOP-2ac:rev.B
- WOP-2ac:rev.C

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP

**Пример**

```
wlc(config-wlc-ap)# ap-model WEP-1L
```

**description**

Данная команда добавляет описание для блоков конфигурации.

Использование отрицательной формы команды (no) удаляет описание.

**Синтаксис**

```
description <DESCRIPTION>
[no] description
```

**Параметры**

<DESCRIPTION> – произвольное описание задается строкой до 255 символов.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-AIRTUNE-PROFILE

CONFIG-WLC-AP

CONFIG-WLC-AP-PROFILE

CONFIG-WLC-IP-POOL

CONFIG-WLC-AP-LOCATION

CONFIG-WLC-PORTAL-PROFILE

CONFIG-WLC-RADIUS-PROFILE

CONFIG-WLC-SSID-PROFILE

CONFIG-WLC-RADIO-2G-PROFILE

CONFIG-WLC-RADIO-5G-PROFILE

CONFIG-WLC-WIDS-PROFILE

**Пример**

```
wlc(config-wlc-ip-pool)# description default_test_pool
```

**hostname**

Данная команда добавляет имя устройству.

Использование отрицательной формы команды (no) удаляет имя.

**Синтаксис**

hostname <NAME>

no hostname

**Параметры**

<NAME> – произвольное имя устройства, задается строкой до 63 символов (латинские буквы, цифры, символы "-" и ".").

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10



## Командный режим

CONFIG-WLC-AP

### Пример

```
wlc(config-wlc-ap)# hostname ap.03-05
```

### override ap-profile

Данная команда переопределяет профиль общих настроек для конкретной точки доступа. Использование отрицательной формы команды (no) удаляет назначенный профиль.

### Синтаксис

```
override ap-profile <PROFILE-NAME>  
no override ap-profile
```

### Параметры

<PROFILE-NAME> – название существующего профиля общих настроек ap-profile, задается строкой до 235 символов.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP

### Пример

```
wlc(config-wlc-ap)# override ap-profile common
```

### override 2g

Данная команда позволяет перейти в меню переопределения параметров радиоинтерфейса, работающего в частотном диапазоне 2,4 ГГц для конкретной точки доступа.

Использование отрицательной формы команды (no) удаляет установленные настройки.

### Синтаксис

```
[no] override 2g
```

### Параметры

Отсутствуют.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP

**Пример**

```
wlc(config-wlc-ap)# override 2g
```

**override 5g**

Данная команда позволяет перейти в меню переопределения параметров радиointерфейса, работающего в частотном диапазоне 5 ГГц для конкретной точки доступа.

Использование отрицательной формы команды (no) удаляет назначенные настройки.

**Синтаксис**

```
[no] override 5g
```

**Параметры**

Отсутствуют.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP

**Пример**

```
wlc(config-wlc-ap)# override 5g
```

**bandwidth**

Данная команда устанавливает ширину канала для радиointерфейса точки доступа.

Использование отрицательной формы команды (no) удаляет заданное значение ширины канала.

**Синтаксис**

```
bandwidth { 20 | 40L | 40U | 80 }
no bandwidth
```

**Параметры**

Отсутствует.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

```
CONFIG-WLC-RADIO-2G-PROFILE
CONFIG-WLC-RADIO-5G-PROFILE
CONFIG-WLC-RADIO-2G-OVERRIDE
CONFIG-WLC-RADIO-5G-OVERRIDE
```

**Пример**

```
wlc(config-wlc-radio-5g-override)# bandwidth 20
```

**channel**

Данная команда устанавливает статический канал для радиointерфейса.

Использование отрицательной формы команды (no) удаляет статический канал. При отсутствии статического канала в индивидуальном профиле будет работать автовыбор каналов из списка limit-channels, заданного в общем профиле настроек radio-2(5)g-profile или переопределенного в индивидуальном профиле точки доступа.

Одновременное использование параметров channel и limit-channels в индивидуальном профиле невозможно.

**Синтаксис**

```
channel <CHANNEL>
no channel
```

**Параметры**

<CHANNEL> – номер используемого канала, доступные значения:

Для диапазона 2g:

- [1.. 13];

Для диапазона 5g:

- 36;
- 40;
- 44;

- 48;
- 52;
- 56;
- 60;
- 64;
- 132;
- 136;
- 140;
- 144;
- 149;
- 153;
- 157;
- 161;
- 165.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-RADIO-2G-PROFILE

CONFIG-WLC-RADIO-5G-PROFILE

CONFIG-WLC-RADIO-2G-OVERRIDE

CONFIG-WLC-RADIO-5G-OVERRIDE

### Пример

```
wlc(config-wlc-radio-5g-override)# channel 36
```

### limit-channels

Данная команда создает ограниченный список каналов для работы автовыбора каналов.

Использование отрицательной формы команды (no) удаляет элемент списка или список целиком.

При заданном параметре limit-channels на точке доступа будет работать автовыбор канала из данного списка. limit-channels по умолчанию определен в общем профиле radio-2(5)g-profile и может быть переопределен для конкретной точки в индивидуальном профиле. При заданном параметре channel на точке доступа будет установлен статический канал и автовыбор будет выключен. Одновременное использование параметров channel и limit-channels в индивидуальном профиле невозможно.

### Синтаксис

```
limit-channels <CHANNEL>[,<CHANNEL>]
no limit-channels { <CHANNEL>[,<CHANNEL>] | all }
```

### Параметры

<CHANNEL> – номер используемого канала, перечисление нескольких каналов осуществляется через запятую без пробелов, доступные значения:

Для диапазона 2g:

- [1.. 13];

Для диапазона 5g:

- 36;
- 40;
- 44;
- 48;
- 52;
- 56;
- 60;
- 64;
- 132;
- 136;
- 140;
- 144;
- 149;
- 153;
- 157;
- 161;
- 165.

all – команда удаляет все добавленные каналы.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-RADIO-2G-OVERRIDE

CONFIG-WLC-RADIO-5G-OVERRIDE

### Пример

```
wlc(config-wlc-radio-5g-override)# limit-channels 36,40,44,48
```

### rates-supported

Данная команда задает возможный набор канальных скоростей для передачи данных в радиозфире при работе по стандартам 802.11 a/b/g в рамках выбранного частотного диапазона.

Использование отрицательной формы команды (no) удаляет установленное значение.

### Синтаксис

```
rates-supported <RATE>[,<RATE>]
```

```
no rates-supported
```

## Параметры

<RATE> – значение канальной скорости, перечисление нескольких значений осуществляется через запятую без пробелов, доступные значения:

Для диапазона 2,4 ГГц:

- 1
- 2
- 5
- 6
- 9
- 11
- 12
- 18
- 24
- 36
- 48
- 54

Для диапазона 5 ГГц:

- 6
- 9
- 12
- 18
- 24
- 36
- 48
- 54

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-RADIO-2G-OVERRIDE

CONFIG-WLC-RADIO-5G-OVERRIDE

## Пример

```
wlc(config-wlc-radio-2g-override)# rates-supported 1,2,5
```

## rates-basic

Данная команда задает возможный набор канальных скоростей для передачи управляющих кадров в радиоэфире в рамках выбранного частотного диапазона.

Использование отрицательной формы команды (no) удаляет установленное значение.

## Синтаксис

```
rates-basic <RATE>[,<RATE>]  
no rates-basic
```

## Параметры

<RATE> – значение канальной скорости, перечисление нескольких значений осуществляется через запятую без пробелов, доступные значения:

Для диапазона 2,4 ГГц:

- 1
- 2
- 5
- 6
- 9
- 11
- 12
- 18
- 24
- 36
- 48
- 54

Для диапазона 5 ГГц:

- 6
- 9
- 12
- 18
- 24
- 36
- 48
- 54

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

10

## Командный режим

```
CONFIG-WLC-RADIO-2G-OVERRIDE
```

```
CONFIG-WLC-RADIO-5G-OVERRIDE
```

## Пример

```
wlc(config-wlc-radio-2g-override)# rates-basic 1,2,5
```

## tx-power

Данная команда устанавливает уровень мощности для радиоинтерфейса.

Использование отрицательной формы команды (no) удаляет значение параметра.

**Синтаксис**

```
tx-power <POWER>
no tx-power
```

**Параметры**

<POWER> – значение мощности сигнала, допустимые значения [0-19], в зависимости от модели точки доступа может принимать значения:

Модель	2,4 ГГц		5 ГГц	
	minimal	maximal	minimal	maximal
WEP-1L	3	16	11	19
WEP-2L	3	16	11	19
WEP-3L	11	16	11	19
WEP-200L	4	16	8	19
WEP-30L	0	16	0	19
WEP-30L-Z	0	16	0	19
WEP-3ax	6	16	10	19
WOP-2L	3	16	11	19
WOP-20L	8	16	11	19
WOP-30L	0	16	0	19
WOP-30LS	0	11	0	11
WOP-30LI	0	16	0	19
WEP-2ac	5	16	1	19
WEP-2ac Smart	5	16	11	19
WOP-2ac	5	16	1	19
WOP-2ac:rev.B	5	16	1	19
WOP-2ac:rev.C	5	16	1	19

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10



**Командный режим**

CONFIG-WLC-RADIO-2G-OVERRIDE

CONFIG-WLC-RADIO-5G-OVERRIDE

**Пример**

```
wlc(config-wlc-radio-5g-override)# tx-power 19
```

**work-mode**

Данная команда устанавливает режим работы радиоинтерфейса.

Использование отрицательной формы команды (no) удаляет значение параметра.

**Синтаксис**

```
work-mode <WORK-MODE>
```

```
no work-mode
```

**Параметры**

<WORK-MODE> – режим работы, доступные значения в зависимости от модели точки доступа:

WEP-3ax, WEP-30L, WEP-30L-Z, WOP-30L, WOP-30LS, WOP-30LI:

- bgn, ax, bgnaх – для диапазона 2g;
- апас, апасах, ах – для диапазона 5g.

WEP-3L:

- n, bg, bgn – для диапазона 2g;
- апас, апасах, ах – для диапазона 5g.

WEP-1L, WEP-2L, WOP-2L, WOP-20L, WEP-200L:

- n, bg, bgn – для диапазона 2g;
- a, ап, апас – для диапазона 5g.

WEP-2ac, WEP-2ac Smart, WOP-2ac, WOP-2ac rev.B, WOP-2ac rev.C:

- n, bg, bgn – для диапазона 2g;
- a, нас, апас – для диапазона 5g.

**Значение по умолчанию**

Отсутствуют.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-RADIO-2G-OVERRIDE

CONFIG-WLC-RADIO-5G-OVERRIDE

**Пример**

```
wlc(config-wlc-radio-5g-override)# work-mode anacax
```

**override wids**

Данная команда позволяет перейти в меню переопределения параметров WIDS для конкретной точки доступа.

Использование отрицательной формы команды (no) удаляет назначенные настройки.

**Синтаксис**

```
[no] override wids
```

**Параметры**

Отсутствуют.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP

**Пример**

```
wlc(config-wlc-ap)# override wids
```

**shared-key**

Данная команда задает общий ключ WIDS индивидуально на точке доступа. Он используется для отслеживания "доверенных" точек доступа в радиоэфире. Если ключ задан одновременно в общих настройках WIDS и/или в локации, и в индивидуальном профиле точки доступа – будет использован ключ из индивидуального профиля.

Использование отрицательной формы команды (no) удаляет ключ.

**Синтаксис**

```
shared-key ascii-text { <CLEAR-TEXT> | encrypted <HASH_SHA512> }
no shared-key
```

**Параметры**

<CLEAR-TEXT> – ключ, задается строкой [10-32] символов;

<HASH\_SHA512> – хеш ключа по алгоритму sha512, задается строкой [20-64] символов.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-WIDS-OVERRIDE

**Пример**

```
wlc(config-wlc-ap-wids-override)# shared-key ascii-text 0123456789
```

**wids-profile**

Данная команда задает профиль настроек WIDS индивидуально для точки доступа.

Использование отрицательной формы команды (no) удаляет установленный профиль WIDS из индивидуального профиля точки доступа.

**Синтаксис**

```
wids-profile <NAME>  
no wids-profile
```

**Параметры**

<NAME> – название профиля WIDS, задается строкой до 235 символов.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-WIDS-OVERRIDE

**Пример**

```
wlc(config-wlc-ap-wids-override)# wids-profile test
```

**white-list**

Данная команда назначает списки MAC-адресов "доверенных" точек доступа, которые сформированы в [object-group mac](#). "Доверенными" считаются точки доступа, которые установлены и управляются оператором.

Использование отрицательной формы команды (no) отменяет использование белого списка.

Списки MAC-адресов могут быть заданы на трех уровнях: в общих настройках WIDS сервиса, в локации и в индивидуальном профиле точки доступа. Приоритет применения списков следующий: настройки индивидуального профиля, настройки локации, общие настройки сервиса. В случае задания на одном уровне только списка одного вида (белого или черного), противоположный список будет использован со следующего уровня. Например, если в индивидуальном профиле точки доступа задан только белый список, то черный будет взят из настроек локации. Если он не задан в локации, то будет взят из общих настроек сервиса. Если требуется задать пустой список, то необходимо создать пустую группу MAC-адресов и использовать ее на нужном уровне.

### Синтаксис

```
white-list <OBJECT-GROUP-MAC>
no white-list
```

### Параметры

<OBJECT-GROUP-MAC> – название существующей группы MAC-адресов, задается строкой до 31 символа.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-AP-WIDS-OVERRIDE

### Пример

```
wlc(config-wlc-ap-wids-override)# white-list test
```

### black-list

Данная команда назначает списки MAC-адресов "вражеских" точек доступа, которые сформированы в [object-group mac](#). "Вражескими" считаются точки, которые несут угрозу для остальных точек доступа в сети, как правило, это известные ТД, которые были обнаружены имитирующими MAC-адрес или SSID исходной ТД.

Использование отрицательной формы команды (no) отменяет использование черного списка.

Списки MAC-адресов могут быть заданы на трех уровнях: в общих настройках WIDS сервиса, в локации и в индивидуальном профиле точки доступа. Приоритет применения списков следующий: настройки индивидуального профиля, настройки локации, общие настройки сервиса. В случае задания на одном уровне только списка одного вида (белого или черного), противоположный список будет использован со следующего уровня. Например, если в индивидуальном профиле точки доступа задан только черный список, то белый будет взят из настроек локации. Если он не задан в локации, то будет взят из общих настроек сервиса. Если требуется задать пустой список, то необходимо создать пустую группу MAC-адресов и использовать ее на нужном уровне.

### Синтаксис

```
black-list <OBJECT-GROUP-MAC>
no black-list
```

## Параметры

<OBJECT-GROUP-MAC> – название существующей группы MAC-адресов, задается строкой до 31 символа.

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-WIDS-OVERRIDE

## Пример

```
wlc(config-wlc-ap-wids-override)# black-list test
```

## wids-disable

Данная команда предназначена для отключения сервиса WIDS на конкретной точке доступа. Включение сервиса выполняется только в общих настройках WIDS, при этом он может быть выключен в каких-то локациях или индивидуально на точках доступа.

Использование отрицательной формы команды (no) не отключает сервис на точке доступа. То есть команда 'no wids-disable' означает, что сервис на точке доступа включен, только если одновременно с этим он включен в общих настройках WIDS. Если в общих настройках сервис выключен, то он не работает на всех точках доступа контроллера и включить его на отдельной точке доступа нельзя.

## Синтаксис

```
[no] wids-disable
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Включено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP

## Пример

```
wlc(config-wlc-ap)# wids-disable
```

## Настройка профиля локации

### ap-location

Данная команда создает/назначает профиль локации, который описывает используемое оборудование и SSID.

Использование отрицательной формы команды (no) удаляет профиль.

### Синтаксис

```
ap-location <NAME>
no ap-location { <NAME> | all }
```

### Параметры

<NAME> – название профиля локации, задается строкой до 235 символов;

all – команда удаляет все созданные профили.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC

CONFIG-WLC-AP

CONFIG- WLC-IP-POOL

### Пример

```
wlc(config-wlc)# ap-location default
```

### airtune-profile

Данная команда позволяет перейти в настройки профиля Airtune.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
airtune-profile <NAME>
no airtune-profile { <NAME> | all }
```

### Параметры

<NAME> – название профиля, задается строкой до 235 символов;

all – параметр для удаления всех профилей.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC

**Пример**

```
wlc(config-wlc)# airtune-profile test
wlc(config-airtune-profile)#
```

**ap-profile**

Данная команда создает профиль общих настроек точек доступа всех типов в выбранной локации. Использование отрицательной формы команды (no) удаляет профиль настроек.

**Синтаксис**

```
ap-profile <NAME>
no ap-profile { <NAME> | all }
```

**Параметры**

<NAME> – название профиля, задается строкой до 235 символов;  
all – команда удаляет все добавленные профили настроек.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC

**Пример**

```
wlc(config-wlc)# ap-profile test
```

**description**

Данная команда добавляет описание для блоков конфигурации. Использование отрицательной формы команды (no) удаляет описание.

**Синтаксис**

```
description <DESCRIPTION>
[no] description
```

**Параметры**

<DESCRIPTION> – произвольное описание задается строкой до 255 символов.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

```
CONFIG-AIRTUNE-PROFILE
CONFIG-WLC-AP
CONFIG-WLC-AP-PROFILE
CONFIG-WLC-AP-POOL
CONFIG-WLC-AP-LOCATION
CONFIG-WLC-PORTAL-PROFILE
CONFIG-WLC-RADIUS-PROFILE
CONFIG-WLC-SSID-PROFILE
CONFIG-WLC-WIDS-PROFILE
```

**Пример**

```
wlc(config-wlc-ip-pool)# description default_test_pool
```

**mode tunnel**

Данная команда включает поднятие туннелей SoftGRE для ТД находящихся в этой локации. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
mode tunnel
no mode tunnel
```

**Параметры**

Отсутствует.

**Значение по умолчанию**

no mode tunnel



**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-LOCATION

**Пример**

```
wlc(config-wlc-ap-location)# mode tunnel
```

**radio-2g-profile**

Данная команда назначает профиль радиointерфейса, работающего в частотном диапазоне 2,4 ГГц. Использование отрицательной формы команды (no) удаляет заданный профиль.

**Синтаксис**

```
radio-2g-profile <PROFILE-ID>  
no radio-2g-profile
```

**Параметры**

<PROFILE-ID> – идентификатор профиля, задается строкой до 235 символов.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC

CONFIG-WLC-AP-LOCATON

**Пример**

```
wlc(config-wlc-ap-location)# radio-2g-profiles default_2g
```

**radio-5g-profile**

Данная команда назначает профиль радиointерфейса, работающего в частотном диапазоне 5 ГГц. Использование отрицательной формы команды (no) удаляет заданный профиль.

**Синтаксис**

```
radio-5g-profile <PROFILE-ID>  
no radio-5g-profile
```

## Параметры

<PROFILE-ID> – идентификатор профиля, задается строкой до 235 символов.

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC

CONFIG-WLC-AP-LOCATON

## Пример

```
wlc(config-wlc-ap-location)# radio-5g-profiles default_5g
```

## ssid-profile

Данная команда создает/назначает профиль SSID.

Использование отрицательной формы команды (no) удаляет SSID.

## Синтаксис

```
ssid-profile <NAME>
```

```
no ssid-profile { <NAME> | all }
```

## Параметры

<NAME> – название профиля SSID, задается строкой до 235 символов;

all – команда удаляет все добавленные профили SSID.

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC

CONFIG-WLC-AP-LOCATION

CONFIG-WLC-SSID-PROFILE

## Пример

```
wlc(config-wlc-ap-location)# ssid-profile test_ssid
```

## timezone

Данная команда устанавливает часовой пояс для выбранной локации. Оптимизация Airtune и обновление по расписанию ТД будут запускаться с учетом часового пояса.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
timezone gmт <TIMEZONE>  
[no] timezone
```

### Параметры

<TIMEZONE> – часовой пояс, допустимые значения от -12 до +12.

### Значение по умолчанию

Часовой пояс установлен из конфигурации устройства.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-RADIUS-PROFILE

### Пример

```
wlc(config-wlc-ap-location)# timezone gmт +7
```

## wids

Данная команда осуществляет переход к настройкам WIDS в локации.

Использование отрицательной формы команды (no) удаляет все сконфигурированные настройки WIDS в локации.

### Синтаксис

```
[no] wids
```

### Параметры

Отсутствуют.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

**Командный режим**

CONFIG-WLC-AP-LOCATION-WIDS

**Пример**

```
wlc(config-wlc-ap-location)# wids
```

**shared-key**

Данная команда задает общий ключ WIDS для точек доступа в одной локации. Он используется для отслеживания доверенных точек доступа в радиозэфире. Если ключ задан одновременно в общих настройках WIDS и в локации – будет использован ключ из локации.

Использование отрицательной формы команды (no) удаляет ключ.

**Синтаксис**

```
shared-key ascii-text { <CLEAR-TEXT> | encrypted <HASH_SHA512> }
no shared-key
```

**Параметры**

<CLEAR-TEXT> – ключ, задается строкой [10-32] символов;

<HASH\_SHA512> – хеш ключа по алгоритму sha512, задается строкой [20-64] символов.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-LOCATION-WIDS

**Пример**

```
wlc(config-wlc-ap-location-wids)# shared-key ascii-text 0123456789
```

**wids-profile**

Данная команда задает профиль настроек WIDS для локации.

Использование отрицательной формы команды (no) удаляет установленный профиль WIDS из локации.

**Синтаксис**

```
wids-profile <NAME>
no wids-profile
```

**Параметры**

<NAME> – название профиля WIDS, задается строкой до 235 символов.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-LOCATION-WIDS

**Пример**

```
wlc(config-wlc-ap-location-wids)# wids-profile test
```

**white-list**

Данная команда назначает списки MAC-адресов "доверенных" точек доступа, которые сформированы в [object-group mac](#). "Доверенными" считаются точки доступа, которые установлены и управляются оператором.

Использование отрицательной формы команды (no) отменяет использование белого списка.

Списки MAC-адресов могут быть заданы на трех уровнях: в общих настройках WIDS сервиса, в локации и в индивидуальном профиле точки доступа. Приоритет применения списков следующий: настройки индивидуального профиля, настройки локации, общие настройки сервиса. В случае задания на одном уровне только списка одного вида (белого или черного), противоположный список будет использован со следующего уровня. Например, если в индивидуальном профиле точки доступа задан только белый список, то черный будет взят из настроек локации. Если он не задан в локации, то будет взят из общих настроек сервиса. Если требуется задать пустой список, то необходимо создать пустую группу MAC-адресов и использовать ее на нужном уровне.

**Синтаксис**

```
white-list <OBJECT-GROUP-MAC>
no white-list
```

**Параметры**

<OBJECT-GROUP-MAC> – название существующей группы MAC-адресов, задается строкой до 31 символа.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-LOCATION-WIDS

## Пример

```
wlc(config-wlc-ap-location-wids)# white-list test
```

## black-list

Данная команда назначает списки MAC-адресов "вражеских" точек доступа, которые сформированы в [object-group mac](#). "Вражескими" считаются точки, которые несут угрозу для остальных точек доступа в сети, как правило, это известные ТД, которые были обнаружены имитирующими MAC-адрес или SSID исходной ТД.

Использование отрицательной формы команды (no) отменяет использование черного списка.

Списки MAC-адресов могут быть заданы на трех уровнях: в общих настройках WIDS сервиса, в локации и в индивидуальном профиле точки доступа. Приоритет применения списков следующий: настройки индивидуального профиля, настройки локации, общие настройки сервиса. В случае задания на одном уровне только списка одного вида (белого или черного), противоположный список будет использован со следующего уровня. Например, если в индивидуальном профиле точки доступа задан только черный список, то белый будет взят из настроек локации. Если он не задан в локации, то будет взят из общих настроек сервиса. Если требуется задать пустой список, то необходимо создать пустую группу MAC-адресов и использовать ее на нужном уровне.

## Синтаксис

```
black-list <OBJECT-GROUP-MAC>
no black-list
```

## Параметры

<OBJECT-GROUP-MAC> – название существующей группы MAC-адресов, задается строкой до 31 символа.

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-LOCATION-WIDS

## Пример

```
wlc(config-wlc-ap-location-wids)# black-list test
```

## wids-disable

Данная команда предназначена для отключения сервиса WIDS в локации. Включение сервиса выполняется только в общих настройках WIDS, при этом он может быть выключен в каких-то локациях или индивидуально на точках доступа.

Использование отрицательной формы команды (no) не отключает сервис в локации. То есть команда 'no wids-disable' означает, что сервис в локации включен, только если одновременно с этим он включен в

общих настройках WIDS. Если в общих настройках сервис выключен, то он не работает на всех точках доступа контроллера и включить его в отдельной локации нельзя.

### Синтаксис

```
[no] wids-disable
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Включено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-AP-LOCATION-WIDS

### Пример

```
wlc(config-wlc-ap-location-wids)# wids-disable
```

## Настройка профиля общих настроек точек доступа

### ap-profile

Данная команда создает профиль общих настроек точек доступа всех типов в выбранной локации. Использование отрицательной формы команды (no) удаляет профиль настроек.

### Синтаксис

```
ap-profile <NAME>  
no ap-profile { <NAME> | all }
```

### Параметры

<NAME> – название профиля, задается строкой до 235 символов;

all – команда удаляет все добавленные профили настроек.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC

### Пример

```
wlc(config-wlc)# ap-profile test
```

## **captive-portal**

Данная команда позволяет перейти в раздел настроек порталной авторизации.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию для дочерних параметров.

### Синтаксис

```
[no] captive-portal
```

### Параметры

Отсутствуют.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE

### Пример

```
wlc(config-wlc-ap-profile)# captive-portal
```

## **ap-ip-alias**

Данная команда задает доменное имя, на которое будет совершаться перенаправление клиентов.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ap-ip-alias <URL>  
no ap-ip-alias
```

### Параметры

<URL> – URL, задается строкой до 255 символов.



**Значение по умолчанию**

redirect.loc

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-CAPTIVE-PORTAL

**Пример**

```
wlc(config-wlc-ap-profile-captive-portal)# ap-ip-alias eltex-co.ru
```

**web-redirector**

Данная команда позволяет изменить названия параметров, используемых в URL авторизации от портала, чтобы точка доступа могла их корректно обработать для прохождения авторизации через RADIUS.

Использование отрицательной формы команды (no) удаляет установленные значения для всех параметров или только для выбранного.

**Синтаксис**

```
web-redirector {error-url | original-url | password | username } <NEW-PARAMETER-NAME>
no web-redirector [error-url | original-url | password | username]
```

**Параметры**

error-url – задает новое название для параметра, содержащего URL, куда будет переадресован клиент в случае ошибки авторизации. Значение по умолчанию на ТД – error\_url.

original-url – задает новое название параметра, содержащего исходный URL, запрошенный клиентом. Клиент будет переадресован на данный URL в случае успешной авторизации. Значение по умолчанию на ТД – original\_url.

password – задает новое название параметра, содержащего пароль для клиента. Значение по умолчанию на ТД – password.

username – задает новое название параметра, содержащего пароль для клиента. Значение по умолчанию на ТД – username.

<NEW-PARAMETER-NAME> – новое название для выбранного параметра.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-CAPTIVE-PORTAL

**Пример**

```
wlc(config-wlc-ap-profile-captive-portal)# web-redirector error-url error_url
```

**description**

Данная команда добавляет описание для блоков конфигурации.

Использование отрицательной формы команды (no) удаляет описание.

**Синтаксис**

```
description <DESCRIPTION>
[no] description
```

**Параметры**

<DESCRIPTION> – произвольное описание задается строкой до 255 символов.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-AIRTUNE-PROFILE

CONFIG-WLC-AP

CONFIG-WLC-AP-PROFILE

CONFIG-WLC-IP-POOL

CONFIG-WLC-AP-LOCATION

CONFIG-WLC-PORTAL-PROFILE

CONFIG-WLC-RADIUS-PROFILE

CONFIG-WLC-SSID-PROFILE

CONFIG-WLC-WIDS-PROFILE

**Пример**

```
wlc(config-wlc-ip-pool)# description default_test_pool
```

**password**

Данная команда задает пароль для управления точкой доступа. Пароль может быть задан как в открытом виде, так и в виде хеш sha512.

Использование отрицательной формы команды (no) удаляет пароль.

**Синтаксис**

```
password { ascii-text <CLEAR-TEXT> | encrypted <HASH_SHA512> }
no password
```

**Параметры**

<CLEAR-TEXT> – пароль, задается строкой [8-64] символов;

<HASH\_SHA512> – хеш пароля по алгоритму sha512, задается строкой [16-128] символов.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE

**Пример**

```
wlc(config-wlc-ap-profile)# password password
```

**services**

Данная команда позволяет перейти в раздел настроек сервисов snmp, ssh, telnet, http(s) для точек доступа.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию для сервисов.

**Синтаксис**

```
[no] services
```

**Параметры**

Отсутствуют.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE

## Пример

```
wlc(config-wlc-ap-profile)# services
```

### ip http port

Данная команда задает порт для подключения к web-конфигуратору по http.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ip http port <PORT>  
no ip http port
```

### Параметры

<PORT> – номер порта для подключения в web-интерфейсу точки доступа, принимает значения [80, 1025...65535].

### Значение по умолчанию

80

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-AP-PROFILE-SERVICES

## Пример

```
wlc(config-wlc-ap-profile-services)# ip http port 80
```

### ip http service

Данная команда включает возможность подключения к web-конфигуратору для http и https.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
[no] ip http service
```

### Параметры

Отсутствуют.

### Значение по умолчанию

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-SERVICES

**Пример**

```
wlc(config-wlc-ap-profile-services)# ip http service
```

**ip https port**

Данная команда задает порт для подключения к web-конфигуратору по https.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
ip https port <PORT>  
no ip http port
```

**Параметры**

<PORT> – номер порта для подключения в web-интерфейсу точки доступа, принимает значения [443, 1025...65535].

**Значение по умолчанию**

443

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-SERVICES

**Пример**

```
wlc(config-wlc-ap-profile-services)# ip https port 443
```

**ip ssh port**

Данная команда задает порт для подключения к точке доступа через протокол ssh.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
ip ssh port <PORT>  
no ip ssh port
```

## Параметры

<PORT> – номер порта для подключения к web-интерфейсу точки доступа, принимает значения [1..65535].

## Значение по умолчанию

22

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE-SERVICES

## Пример

```
wlc(config-wlc-ap-profile-services)# ip ssh port 22
```

## ip ssh service

Данная команда включает возможность подключения к точке доступа через протокол ssh. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
[no] ip ssh service
```

## Параметры

Отсутствуют.

## Значение по умолчанию

Отключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE-SERVICES

## Пример

```
wlc(config-wlc-ap-profile-services)# ip ssh service
```

## ip telnet port

Данная команда задает порт для подключения к точке доступа через протокол telnet. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
ip telnet port <PORT>  
no ip telnet port
```

**Параметры**

<PORT> – номер порта для подключения к web-интерфейсу точки доступа, принимает значения [1..65535].

**Значение по умолчанию**

23

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-SERVICES

**Пример**

```
wlc(config-wlc-ap-profile-services)# ip telnet port 23
```

**ip telnet service**

Данная команда включает возможность подключения к точке доступа через протокол ssh. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
[no] ip telnet service
```

**Параметры**

Отсутствуют.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-SERVICES

**Пример**

```
wlc(config-wlc-ap-profile-services)# ip telnet service
```

**snmp-server**

Данная команда включает возможность работы по протоколу SNMP.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
[no] snmp-server
```

**Параметры**

Отсутствуют.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-SERVICES

**Пример**

```
wlc(config-wlc-ap-profile-services)# snmp-server
```

**snmp-server community**

Данная команда назначает community для доступа по протоколу SNMP.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
snmp-server community <NAME> {ro | rw}
```

```
no snmp-server community {ro | rw}
```

**Параметры**

<NAME> – имя community для работы с трапами;

ro – режим "только для чтения";

rw – режим "чтение и запись".

**Значение по умолчанию**

```
snmp-server community public ro
```

```
snmp-server community private rw
```

**Необходимый уровень привилегий**

10



## Командный режим

CONFIG-WLC-AP-PROFILE-SERVICES

### Пример

```
wlc(config-wlc-ap-profile-services)# snmp-server community public ro
```

### snmp-server host

Данная команда назначает хост для работы с трапами.

Использование отрицательной формы команды (no) удаляет настроенный хост.

### Синтаксис

```
snmp-server host <ADDR> {informs | traps <VERSION>}
```

```
no snmp-server host {informs | traps <VERSION>}
```

### Параметры

<ADDR> – IP-адрес RADIUS-сервера, задаётся в виде: AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<VERSION> – версия, возможные значения: 1 и 2с.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE-SERVICES

### Пример

```
wlc(config-wlc-ap-profile-services)# snmp-server host 192.168.1.1 traps 2c
```

### z-wave

Данная команда включает функционал Z-Wave на ТД.

Использование отрицательной формы команды (no) отключает функционал Z-Wave на ТД.

### Синтаксис

```
z-wave
```

```
no z-wave
```

### Параметры

Отсутствует.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-SERVICES

**Пример**

```
wlc(config-wlc-ap-profile-services)# z-wave
```

**z-wave address**

Данная команда устанавливает адрес сервера «Eltex Smart Cloud (Eltex SC)».

Использование отрицательной формы команды (no) удаляет заданный адрес.

**Синтаксис**

```
z-wave address { <ADDR> | <URL> }  
no z-wave address
```

**Параметры**

<ADDR> – IP-адрес сервера, задается в виде: AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

<URL> – URL-адрес сервера, задается строкой [1-253] символов.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-SERVICES

**Пример**

```
wlc(config-wlc-ap-profile-services)# z-wave address smart.eltex-co.ru
```

**z-wave port**

Данная команда устанавливает порт сервера «Eltex Smart Cloud (Eltex SC)».

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
z-wave port <PORT>  
no z-wave port
```

**Параметры**

<PORT> – номер порта для подключения к серверу, принимает значения [443,1025-65535].

**Значение по умолчанию**

8072

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-SERVICES

**Пример**

```
wlc(config-wlc-ap-profile-services)# z-wave port 8070
```

**z-wave secure**

Данная команда включает использование SSL-соединения при обмене с сервером.

Использование отрицательной формы команды (no) отключает использование SSL-соединения при обмене с сервером.

**Синтаксис**

```
z-wave secure  
no z-wave secure
```

**Параметры**

Отсутствует.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-SERVICES

**Пример**

```
wlc(config-wlc-ap-profile-services)# z-wave secure
```

## Настройка профилей параметров радио точек доступа

### radio-2g-profile

Данная команда создает профиль настроек радиоинтерфейса, работающего в частотном диапазоне 2,4 ГГц.

Использование отрицательной формы команды (no) удаляет заданный профиль.

#### Синтаксис

```
radio-2g-profile <PROFILE-ID>
no radio-2g-profile { <PROFILE-ID> | all }
```

#### Параметры

<PROFILE-ID> – идентификатор профиля, задается строкой до 235 символов;

all – команда удаляет все добавленные профили.

#### Значение по умолчанию

Отсутствует.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-WLC

CONFIG-WLC-AP-LOCATON

#### Пример

```
wlc(config-wlc)# radio-2g-profiles default_2g
```

### radio-5g-profile

Данная команда создает профиль настроек радиоинтерфейса, работающего в частотном диапазоне 5 ГГц.

Использование отрицательной формы команды (no) удаляет заданный профиль.

#### Синтаксис

```
radio-5g-profile <PROFILE-ID>
no radio-5g-profile { <PROFILE-ID> | all }
```

#### Параметры

<PROFILE-ID> – идентификатор профиля, задается строкой до 235 символов;

all – команда удаляет все добавленные профили.

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC

CONFIG-WLC-AP-LOCATON

## Пример

```
wlc(config-wlc)# radio-5g-profiles default_5g
```

## rates-supported

Данная команда задает возможный набор канальных скоростей для передачи данных в радиозфире при работе по стандартам 802.11 a/b/g в рамках выбранного частотного диапазона.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
rates-supported <RATE>[,<RATE>]
```

```
no rates-supported
```

## Параметры

<RATE> – значение канальной скорости, перечисление нескольких значений осуществляется через запятую без пробелов, доступные значения:

Для диапазона 2,4 ГГц:

- 1
- 2
- 5
- 6
- 9
- 11
- 12
- 18
- 24
- 36
- 48
- 54

Для диапазона 5 ГГц:

- 6
- 9
- 12
- 18
- 24

- 36
- 48
- 54

### Значение по умолчанию

1,2,5,6,9,11,12,18,24,36,48,54 – для частотного диапазона 2,4 ГГц;

6,9,12,18,24,36,48,54 – для частотного диапазона 5 ГГц.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-RADIO-2G-PROFILE

CONFIG-WLC-RADIO-5G-PROFILE

### Пример

```
wlc(config-wlc-radio-2g-profile)# rates-supported 1,2,5
```

### rates-basic

Данная команда задает возможный набор канальных скоростей для передачи управляющих кадров в радиоэфире в рамках выбранного частотного диапазона.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
rates-basic <RATE>[,<RATE>]
no rates-basic
```

### Параметры

<RATE> – значение канальной скорости, перечисление нескольких значений осуществляется через запятую без пробелов, доступные значения:

Для диапазона 2,4 ГГц:

- 1
- 2
- 5
- 6
- 9
- 11
- 12
- 18
- 24
- 36
- 48
- 54

Для диапазона 5 ГГц:

- 6
- 9

- 12
- 18
- 24
- 36
- 48
- 54

### Значение по умолчанию

1,2,5,11 – для частотного диапазона 2,4 ГГц;

6,12,24 – для частотного диапазона 5 ГГц.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-RADIO-2G-PROFILE

CONFIG-WLC-RADIO-5G-PROFILE

### Пример

```
wlc(config-wlc-radio-2g-profile)# rates-basic 1,2,5
```

### bandwidth

Данная команда назначает ширину канала для радиоинтерфейса точки доступа.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
bandwidth { 20 | 40L | 40U | 80 }
no bandwidth
```

### Параметры

Отсутствует.

### Значение по умолчанию

20

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-RADIO-2G-PROFILE

CONFIG-WLC-RADIO-5G-PROFILE

CONFIG-WLC-RADIO-2G-OVERRIDE

CONFIG-WLC-RADIO-5G-OVERRIDE

## Пример

```
wlc(config-wlc-radio-5g-profile)# bandwidth 20
```

## description

Данная команда добавляет описание для блоков конфигурации.

Использование отрицательной формы команды (no) удаляет описание.

## Синтаксис

```
[no] description <DESCRIPTION>
```

## Параметры

<DESCRIPTION> – произвольное описание задается строкой до 255 символов.

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-AIRTUNE-PROFILE

CONFIG-WLC-AP

CONFIG-WLC-AP-PROFILE

CONFIG-WLC-IP-POOL

CONFIG-WLC-AP-LOCATION

CONFIG-WLC-PORTAL-PROFILE

CONFIG-WLC-RADIUS-PROFILE

CONFIG-WLC-SSID-PROFILE

CONFIG-WLC-RADIO-2G-PROFILE

CONFIG-WLC-RADIO-5G-PROFILE

CONFIG-WLC-WIDS-PROFILE

## Пример

```
wlc(config-wlc-radio-5g-profile)# description default
```

## dfs

Данная команда определяет режим динамического выбора частоты. Данный механизм требует от беспроводных устройств сканировать радиоэфир и избегать использования каналов, совпадающих с каналами, на которых работают радиолокационные системы в диапазоне 5 ГГц.



Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
dfs {auto | disabled | forced}
no dfs
```

### Параметры

auto – механизм включен;

disabled – механизм выключен. DFS-каналы не доступны для выбора;

forced – механизм выключен. DFS-каналы доступны для выбора;

### Значение по умолчанию

auto

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-RADIO-5G-PROFILE

### Пример

```
wlc(config-wlc-radio-5g-profile)# dfs forced
```

### ip dhcp information option action

Данная команда позволяет выбрать необходимое действие, применяемое к 82 опции DHCP.

Использование отрицательной формы команды (no) без аргументов устанавливает действие по умолчанию.

### Синтаксис

```
ip dhcp information option {replace | keep | drop}
no ip dhcp information option
```

### Параметры

replace – опция 82 будет изменена/добавлена точкой доступа;

keep – опция 82 не будет обработана. Будет использовано значение опции, указанное в запросах от клиента, если таковое имеется;

drop – опция 82 будет удалена.

### Значение по умолчанию

keep

### Необходимый уровень привилегий

10

**Командный режим**

CONFIG-WLC-RADIO-2G-PROFILE

CONFIG-WLC-RADIO-5G-PROFILE

**Пример**

```
wlc(config-wlc-radio-5g-profile)# ip dhcp information option action replace
```

**ip dhcp information option format-type circuit-id**

Данная команда позволяет выбрать необходимый формат значений CID 82 опции DHCP.

Использование отрицательной формы команды (no) без аргументов определяет значение circuit-id по умолчанию.

При выборе custom дополнительно необходимо указать значение данного поля.

**Синтаксис**

```
ip dhcp information option format-type circuit-id {ap-mac-ssid | ssid | custom <value>}
no ip dhcp information option format-type circuit-id
```

**Параметры**

ap-mac-ssid – значение circuit-id будет содержать <MAC-адрес точки доступа> – <имя SSID>;

ssid – значение circuit-id будет содержать <имя SSID>;

custom <value> – значение circuit-id будет указано произвольное, от 1 до 52 символов.

**Значение по умолчанию**

ap-mac-ssid

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-RADIO-2G-PROFILE

CONFIG-WLC-RADIO-5G-PROFILE

**Примеры**

```
wlc(config-wlc-radio-2g-profile)# ip dhcp information option format-type circuit-id ssid
wlc(config-wlc-radio-5g-profile)# ip dhcp information option format-type circuit-id custom
test-cid-opt82
```

**ip dhcp information option format-type remote-id**

Данная команда позволяет выбрать необходимый формат значений RID 82 опции DHCP.

Использование отрицательной формы команды (no) без аргументов определяет значение remote-id по умолчанию.

При выборе custom дополнительно необходимо указать значение данного поля.

### Синтаксис

```
ip dhcp information option format-type remote-id {custom <value> | ap-domain | ap-mac |
client-mac}
no ip dhcp information option format-type remote-id
```

### Параметры

custom <value> – значение remote-id будет указано произвольное, от 1 до 63 символов;  
ap-domain – значение remote-id будет содержать <имя домена>, в котором находится точка доступа;  
ap-mac – значение remote-id будет содержать <MAC-адрес точки доступа>;  
client-mac – значение remote-id будет содержать <MAC-адрес клиента>.

### Значение по умолчанию

client mac

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-RADIO-2G-PROFILE

CONFIG-WLC-RADIO-5G-PROFILE

### Примеры

```
wlc(config-wlc-radio-2g-profile)# ip dhcp information option format-type remote-id ap-mac
wlc(config-wlc-radio-5g-profile)# ip dhcp information option format-type circuit-id custom
test-rid-opt82
```

### ip dhcp information option format-type mac-address

Данная команда позволяет выбрать необходимый формат передаваемого MAC-адреса.

Использование отрицательной формы команды (no) без аргументов определяет формат MAC-адреса по умолчанию.

### Синтаксис

```
ip dhcp information option format-type mac-address {default | radius}
no ip dhcp information option format-type remote-id
```

### Параметры

default – MAC-адрес будет передаваться с ":" разделителем;  
radius – MAC-адрес будет передаваться с "-" разделителем.

**Значение по умолчанию**

default

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-RADIO-2G-PROFILE

CONFIG-WLC-RADIO-5G-PROFILE

**Примеры**

```
wlc(config-wlc-radio-2g-profile)# ip dhcp information option format-type mac-address radius
```

**limit-channels**

Данная команда создает ограниченный список каналов для работы автовыбора каналов.

Использование отрицательной формы команды (no) без аргументов – устанавливает список каналов по умолчанию.

При заданном параметре limit-channels на точке доступа будет работать автовыбор канала из данного списка. limit-channels по умолчанию определен в общем профиле radio-2(5)g-profile и может быть переопределен для конкретной точки в индивидуальном профиле. При заданном параметре channel в индивидуальном профиле на точке доступа будет установлен статический канал и автовыбор будет выключен.

**Синтаксис**

```
limit-channels <CHANNEL>[,<CHANNEL>]
no limit-channels
```

**Параметры**

<CHANNEL> – номер используемого канала, перечисление нескольких каналов осуществляется через запятую без пробелов, доступные значения:

Для диапазона 2g:

- [1.. 13];

Для диапазона 5g:

- 36;
- 40;
- 44;
- 48;
- 52;
- 56;
- 60;
- 64;
- 132;
- 136;
- 140;
- 144;

- 149;
- 153;
- 157;
- 161;
- 165.

all – восстанавливает список каналов по умолчанию.

### Значение по умолчанию

1,6,11 – для частотного диапазона 2,4 ГГц;

36,40,44,48 – для частотного диапазона 5 ГГц.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-RADIO-2G-PROFILE

CONFIG-WLC-RADIO-5G-PROFILE

### Пример

```
wlc(config-wlc-radio-2g-profile)# limit-channels 1,6,11
```

### obss-coexistence

Данная команда отвечает за работу режима автоматического уменьшения ширины канала при загруженном радиоэфире.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
obss-coexistence {on | off}
no obss-coexistence
```

### Параметры

on – режим автоматического уменьшения ширины канала активирован;

off – режим автоматического уменьшения ширины канала выключен.

### Значение по умолчанию

off

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-RADIO-2G-PROFILE

CONFIG-WLC-RADIO-5G-PROFILE

## Пример

```
wlc(config-wlc-radio-5g-profile)# obs-coexistence off
```

## tx-power

Данная команда устанавливает уровень мощности для радиointерфейса.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
tx-power {minimal | low | middle | high | maximal}
no tx-power
```

## Параметры

Возможные значения параметра в зависимости от модели точки доступа устанавливают следующие значения мощности в дБм:

Модель	2,4 ГГц					5 ГГц				
	minimal	low	middle	high	maximal	minimal	low	middle	high	maximal
WEP-1L	3	6	10	13	16	11	13	15	17	19
WEP-2L	3	6	10	13	16	11	13	15	17	19
WEP-3L	11	12	14	15	16	11	13	15	17	19
WEP-200L	4	4	7	10	16	8	11	14	17	19
WEP-30L	0	4	8	12	16	0	5	10	15	19
WEP-30L-Z	0	4	8	12	16	0	5	10	15	19
WEP-3ax	6	8	11	14	16	10	12	15	17	19
WOP-2L	3	6	10	13	16	11	13	15	17	19
WOP-20L	8	10	12	14	16	11	13	15	17	19
WOP-30L	0	4	8	12	16	0	5	10	15	19
WOP-30LS	0	3	6	9	11	0	3	6	9	11
WOP-30LI	0	4	8	12	16	0	5	10	15	19
WEP-2ac	5	8	11	14	16	1	6	10	15	19
WEP-2ac Smart	5	8	11	14	16	11	13	15	17	19
WOP-2ac	5	8	11	14	16	1	6	10	15	19
WOP-2ac:rev.B	5	8	11	14	16	1	6	10	15	19

WOP-2ac:rev.C	5	8	11	14	16	1	6	10	15	19
---------------	---	---	----	----	----	---	---	----	----	----

**Значение по умолчанию**

maximal

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-RADIO-2G-PROFILE

CONFIG-WLC-RADIO-5G-PROFILE

**Пример**

```
wlc(config-wlc-radio-5g-profile)# tx-power maximal
```

**work-mode**

Данная команда устанавливает режим работы радиоинтерфейса. Если точка доступа не поддерживает указанный режим, то выставляется максимально возможный смешанный режим из поддерживаемых. Например, при work-mode апасах на точку доступа WEP-2L установится режим апас, потому что режим ах она не поддерживает.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
work-mode <WORK-MODE>
```

```
no work-mode
```

**Параметры**

<WORK-MODE> – режим работы, доступные значения:

- bg, пах, bgпах – для частотного диапазона 2,4 ГГц;
- апасах – для частотного диапазона 5 ГГц.

**Значение по умолчанию**

bgпах – для частотного диапазона 2,4 ГГц;

апасах – для частотного диапазона 5 ГГц.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-RADIO-2G-PROFILE

CONFIG-WLC-RADIO-5G-PROFILE

## Пример

```
wlc(config-wlc-radio-5g-profile)# work-mode anacax
```

## Настройка адресного пространства

### ip-pool

Данная команда создает адресное пространство со списком разрешенных IP-адресов.

Использование отрицательной формы команды (no) удаляет адресное пространство.

### Синтаксис

```
ip-pool <NAME>  
no ip-pool { <NAME> | all }
```

### Параметры

<NAME> – название адресного пространства, задается строкой до 235 символов;

all – команда удаляет все заданные адресные пространства.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC

## Пример

```
wlc(config-wlc)# ip-pool test
```

### ap-location

Данная команда создает/назначает профиль локации, который описывает используемое оборудование и SSID.

Использование отрицательной формы команды (no) удаляет профиль.

### Синтаксис

```
ap-location <NAME>  
no ap-location { <NAME> | all }
```



**Параметры**

<NAME> – название профиля локации, задается строкой до 235 символов;

all – команда удаляет все созданные профили.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC

CONFIG-WLC-AP

CONFIG-WLC-IP-POOL

**Пример**

```
wlc(config-wlc)# ap-location default
```

**description**

Данная команда добавляет описание для блоков конфигурации.

Использование отрицательной формы команды (no) удаляет описание.

**Синтаксис**

```
[no] description <DESCRIPTION>
```

**Параметры**

<DESCRIPTION> – произвольное описание задается строкой до 255 символов.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-AIRTUNE-PROFILE

CONFIG-WLC-AP

CONFIG-WLC-AP-PROFILE

CONFIG-WLC-IP-POOL

CONFIG-WLC-AP-LOCATION

CONFIG-WLC-PORTAL-PROFILE

CONFIG-WLC-RADIUS-PROFILE

CONFIG-WLC-SSID-PROFILE

CONFIG-WLC-WIDS-PROFILE

**Пример**

```
wlc(config-wlc-ip-pool)# description default_test_pool
```

**network**

Данная команда формирует список разрешенных IP-адресов.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
network <ADDR/LEN>
no network
```

**Параметры**

<ADDR/LEN> – IP-адрес и маска подсети, задается в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].

**Значение по умолчанию**

0.0.0.0/0

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-IP-POOL

**Пример**

```
wlc(config-wlc-ip-pool)# network 0.0.0.0/0
```

**Настройка фильтрации логов****log-filter**

Данная команда активирует фильтрацию отображения логов по одной или группе точек доступа.

Использование отрицательной формы команды (no) отключает функцию.

**Синтаксис**

```
log-filter <OBJECT-GROUP>
[no] log-filter
```

## Параметры

<OBJECT-GROUP> – название object-group, задается строкой до 31 символа, должно совпадать с существующей object-group.

## Значение по умолчанию

Отключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC

## Пример

```
wlc(config-wlc)# log-filter test
```

## Настройка использования портала

### portal-profile

Данная команда настраивает использование портала.

Использование отрицательной формы команды (no) удаляет настройки портала.

### Синтаксис

```
portal-profile <PORTAL-NAME>  
no portal-profile [ <PORTAL-NAME> | all ]
```

## Параметры

<PORTAL-NAME> – название портала, задается строкой до 235 символов;

all – команда удаляет все созданные порталы.

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC

CONFIG-WLC-SSID-PROFILE

## Пример

```
wlc(config-wlc)# portal-profile test
```

## age-timeout

Данная команда устанавливает временной интервал, в течение которого точка доступа "помнит" клиента.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
age-timeout <TIMEOUT>
no age-timeout
```

## Параметры

<TIMEOUT> – временной интервал в секундах, принимает значения от 0 до 604800.

**i** Значение **age-timeout 0** означает, что точка будет "помнить" авторизованного клиента бесконечно, до перезагрузки.

## Значение по умолчанию

86400

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-PORTAL-PROFILE

## Пример

```
wlc(config-wlc-portal-profile)# age-timeout 86400
```

## description

Данная команда добавляет описание для блоков конфигурации.

Использование отрицательной формы команды (no) удаляет описание.

## Синтаксис

```
[no] description <DESCRIPTION>
```

## Параметры

<DESCRIPTION> – произвольное описание задается строкой до 255 символов.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-AIRTUNE-PROFILE

CONFIG-WLC-AP

CONFIG-WLC-AP-PROFILE

CONFIG-WLC-IP-POOL

CONFIG-WLC-AP-LOCATION

CONFIG-WLC-PORTAL-PROFILE

CONFIG-WLC-RADIUS-PROFILE

CONFIG-WLC-SSID-PROFILE

CONFIG-WLC-WIDS-PROFILE

**Пример**

```
wlc(config-wlc-ip-pool)# description default_test_pool
```

**redirect-url**

Данная команда устанавливает URL перенаправления на портал.

Использование отрицательной формы команды (no) удаляет заданный URL.

**Синтаксис**

```
redirect-url <URL>
```

```
no redirect-url
```

**Параметры**

<URL> – URL-перенаправления, задается строкой [4-255] символов. Формирование строки зависит от [verification-mode](#). В режиме portal строка будет сформирована без изменений. URL, заданный в виде: `https://192.168.0.1:8080/eltex_portal/` в режиме external-portal, будет автоматически дополнен следующими параметрами:

```
https://192.168.0.1:8080/eltex_portal/?
switch_url=<SWITCH_URL>&ap_mac=<AP_MAC>&client_mac=<CLIENT_MAC>&wlan=<SSID>&redirect=<ORIGINAL_
URL>
```

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-PORTAL-PROFILE

**Пример**

```
wlc(config-wlc-portal-profile)#redirect-url http://softwlc.eltex.loc:8080/eltex_portal/
```

**redirect-url-custom**

Данная команда устанавливает пользовательский URL перенаправления на портал, в котором можно подставить параметры: <ORIGINAL\_URL>, <SWITCH\_URL>, <NAS\_ID>, <SSID>, <CLIENT\_MAC>,<AP\_MAC>.

Использование отрицательной формы команды (no) удаляет заданный URL.

**Синтаксис**

```
redirect-url-custom <URL>
no redirect-url-custom
```

**Параметры**

<URL> – URL-перенаправления, задается строкой [4-255] символов.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-PORTAL-PROFILE

**Пример**

```
wlc(config-wlc-portal-profile)# redirect-url-custom https://192.168.0.1:8080/eltex_portal/?
switch_url=<SWITCH_URL>&client_mac=<CLIENT_MAC>&ap_mac=<AP_MAC>&wlan=<SSID>
```

**verification-mode**

Данная команда устанавливает режим работы с порталом.

Использование отрицательной формы команды (no) удаляет установленный режим.

**Синтаксис**

```
verification-mode {external-portal | portal}
no verification-mode
```

## Параметры

external-portal – режим интеграции с внешним порталом cisco-like;

portal – режим интеграции с порталом SoftWLC.

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-PORTAL-PROFILE

## Пример

```
wlc(config-wlc-portal-profile)# verification-mode external-portal
```

## virtual-portal-name

Данная команда устанавливает название портала.

Использование отрицательной формы команды (no) удаляет название портала.

## Синтаксис

```
virtual-portal-name <PORTAL-NAME>
```

```
no virtual-portal-name
```

## Параметры

<PORTAL-NAME> – название портала, задается строкой до 235 символов.

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-PORTAL-PROFILE

## Пример

```
wlc(config-wlc-portal-profile)# virtual-portal-name default
```

## white-list

Данная команда назначает белые списки доступа, которые сформированы в [object-group mac](#).

Использование отрицательной формы команды (no) удаляет все белые списки, а также позволяет удалить список по категории: белый список по подсетям или белый список по доменным именам.

### Синтаксис

```
white-list {address | domain} <OBJECT-GROUP>
no white-list {address | domain}
```

### Параметры

address – указывает на использование группы объектов, сформированной по подсетям [object-group networkobject-group network](#);

domain – указывает на использование группы объектов, сформированной по доменным именам [object-group urlobject-group url](#);

<OBJECT-GROUP> – название существующей группы объектов, задается строкой до 31 символа.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-PORTAL-PROFILE

### Пример

```
wlc(config-wlc-portal-profile)# white-list address test
```

## Настройка профиля для RADIUS-сервера

### radius-profile

Данная команда назначает/настраивает использование RADIUS-сервера.

Использование отрицательной формы команды (no) отменяет назначение/настройки RADIUS-сервера.

### Синтаксис

```
radius-profile <RADIUS-ID>
no radius-profile { <RADIUS-ID> | all }
```

### Параметры

<RADIUS-ID> – идентификатор RADIUS-сервера, задается строкой до 235 символов;

all – команда удаляет все созданные RADIUS-сервера.

### Значение по умолчанию

Отсутствует.



**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC

CONFIG-WLC-PORTAL-PROFILE

CONFIG-WLC-SSID-PROFILE

**Пример**

```
wlc(config-wlc)# radius-profile test
```

**acct-address**

Данная команда определяет IP-адрес RADIUS-сервера, используемого для аккаунтинга.

Использование отрицательной формы команды (no) удаляет указанный IP-адрес.

**Синтаксис**

```
acct-address <ADDR>
```

```
no acct-address
```

**Параметры**

<ADDR> – IP-адрес RADIUS-сервера, задается в виде: AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-RADIUS-PROFILE

**Пример**

```
wlc(config-wlc-radius-profile)# acct-address 192.168.1.1
```

**acct-enable**

Данная команда активирует отправку аккаунтинга на RADIUS-сервер.

Использование отрицательной формы команды (no) отключает отправку аккаунтинга.

**Синтаксис**

```
[no] acct-enable
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-RADIUS-PROFILE

**Пример**

```
wlc(config-wlc-radius-profile)# acct-enable
```

**acct-interval**

Данная команда задает временной интервал обновления аккаунтинга.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
acct-interval <TIME>  
no acct-interval
```

**Параметры**

<TIME> – промежуток времени в секундах, через который аккаунтинг будет обновлен, принимает значения [1..86400].

**Значение по умолчанию**

600 секунд.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-RADIUS-PROFILE

**Пример**

```
wlc(config-wlc-radius-profile)# acct-interval 600
```

**acct-password**

Данная команда указывает пароль авторизации для RADIUS-сервера, используемого для аккаунтинга. Пароль может быть задан как в открытом виде, так и в виде хеш sha512.

Использование отрицательной формы команды (no) удаляет заданный пароль.

**Синтаксис**

```
acct-password {ascii-text <CLEAR-TEXT> | encrypted <HASH_SHA512>}
no acct-password
```

**Параметры**

<CLEAR-TEXT> – пароль, задается строкой [8-64] символа.

<HASH\_SHA512> – хеш пароля по алгоритму sha512, задается строкой [16-128] символов.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-RADIUS-PROFILE

**Пример**

```
wlc(config-wlc-radius-profile)# acct-password ascii-text password
```

**acct-periodic**

Данная команда активирует периодическое обновление аккаунтинга.

Использование отрицательной формы команды (no) отключает обновление аккаунтинга.

**Синтаксис**

```
[no] acct-periodic
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

## Командный режим

CONFIG-WLC-RADIUS-PROFILE

### Пример

```
wlc(config-wlc-radius-profile)# acct-periodic
```

### acct-port

Данной командой задается номер порта для обмена данными с удаленным RADIUS-сервером при выполнении аккаунтинга.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
acct-port <PORT>  
no acct-port
```

### Параметры

<PORT> – номер UDP-порта для обмена данными с удаленным сервером, принимает значения [1..65535].

### Значение по умолчанию

1813

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-RADIUS-PROFILE

### Пример

```
wlc(config-wlc-radius-profile)# acct-port 1813
```

### auth-acct-id-send

Данная команда активирует передачу идентификатора сессии в запросах аккаунтинга.

Использование отрицательной формы команды (no) деактивирует функцию.

### Синтаксис

```
[no] auth-acct-id-send
```

### Параметры

Отсутствуют.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-RADIUS-PROFILE

**Пример**

```
wlc(config-wlc-radius-profile)# auth-acct-id-send
```

**auth-address**

Данная команда определяет IP-адрес RADIUS-сервера авторизации.

Использование отрицательной формы команды (no) удаляет заданный адрес.

**Синтаксис**

```
auth-address <ADDR>  
no auth-address
```

**Параметры**

<ADDR> – IP-адрес RADIUS-сервера, задается в виде: AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-RADIUS-PROFILE

**Пример**

```
wlc(config-wlc-radius-profile)# auth-address 192.168.1.1
```

**auth-password**

Данная команда указывает пароль авторизации для RADIUS-сервера, который используется для авторизации и аутентификации. Пароль может быть задан как в открытом виде, так и в виде хеш sha512.

Использование отрицательной формы команды (no) удаляет заданный пароль.

**Синтаксис**

```
auth-password { ascii-text <CLEAR-TEXT> | encrypted <HASH_SHA512> }
no auth-password
```

**Параметры**

<CLEAR-TEXT> – пароль, задается строкой [8-64] символа;

<HASH\_SHA512> – хеш пароля по алгоритму sha512, задается строкой [16-128] символов.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-RADIUS-PROFILE

**Пример**

```
wlc(config-wlc-radius-profile)# auth-password password
```

**auth-port**

Данной командой задается номер порта для обмена данными с удаленным RADIUS-сервером при выполнении аутентификации и авторизации.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
auth-port <PORT>
no auth-port
```

**Параметры**

<PORT> – номер UDP-порта для обмена данными с удаленным сервером, принимает значения [1..65535].

**Значение по умолчанию**

1812

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-RADIUS-PROFILE

## Пример

```
wlc(config-wlc-radius-profile)# auth-port 1812
```

## description

Данная команда добавляет описание для блоков конфигурации.  
Использование отрицательной формы команды (no) удаляет описание.

## Синтаксис

```
[no] description <DESCRIPTION>
```

## Параметры

<DESCRIPTION> – произвольное описание задается строкой до 255 символов.

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-AIRTUNE-PROFILE

CONFIG-WLC-AP

CONFIG-WLC-AP-PROFILE

CONFIG-WLC-IP-POOL

CONFIG-WLC-AP-LOCATION

CONFIG-WLC-PORTAL-PROFILE

CONFIG-WLC-RADIUS-PROFILE

CONFIG-WLC-SSID-PROFILE

CONFIG-WLC-WIDS-PROFILE

## Пример

```
wlc(config-wlc-ip-pool)# description default_test_pool
```

## domain

Данная команда создает/назначает домен.  
Использование отрицательной формы команды (no) удаляет домен.

**Синтаксис**

```
domain <DOMAIN>  
no domain
```

**Параметры**

<DOMAIN> – идентификатор домена, задается строкой до 235 символов.

**Значение по умолчанию**

root

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-RADIUS-PROFILE

**Пример**

```
wlc(config-wlc-radius-profile)# domain root
```

**nas-id**

Данная команда добавляет параметр NAS-ID в состав GET-запроса от клиента.

Использование отрицательной форсы команды (no) удаляет добавление параметра в запрос.

**Синтаксис**

```
nas-id <ID>  
no nas-id
```

**Параметры**

<ID> – номер идентификатора, задается строкой от 1 до 235.

**Значение по умолчанию**

Отсутствует

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-RADIUS-PROFILE

**Пример**

```
wlc(config-wlc-radius-profile)# nas-id 2
```



## tls-enable

Данная команда активирует использование TLS при авторизации.

Использование отрицательной формы команды (no) отключает функцию.

### Синтаксис

```
[no] tls-enable
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-RADIUS-PROFILE

### Пример

```
wlc(config-wlc-radius-profile)# tls-enable
```

## Настройка профиля ограничения скорости

### policy-profile ap

Данная команда создает профиль ограничения скорости средствами точки доступа.

Использование отрицательной формы команды (no) удаляет профиль настроек.

### Синтаксис

```
policy-profile ap <POLICY-NAME>  
no policy-profile ap { <POLICY-NAME> | all }
```

### Параметры

<POLICY-NAME> – название профиля, задается строкой до 235 символов.

all – команда удаляет все созданные профили.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC

### Пример

```
wlc(config-wlc)# policy-profile ap testpolicyprofile
```

### description

Данная команда добавляет описание для блоков конфигурации.  
Использование отрицательной формы команды (no) удаляет описание.

### Синтаксис

```
[no] description <DESCRIPTION>
```

### Параметры

<DESCRIPTION> – произвольное описание задается строкой до 255 символов.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-AIRTUNE-PROFILE

CONFIG-WLC-AP

CONFIG-WLC-AP-PROFILE

CONFIG-WLC-IP-POOL

CONFIG-WLC-AP-LOCATION

CONFIG-WLC-PORTAL-PROFILE

CONFIG-WLC-POLICY-PROFILE-AP

CONFIG-WLC-RADIUS-PROFILE

CONFIG-WLC-SSID-PROFILE

CONFIG-WLC-WIDS-PROFILE

### Пример

```
wlc(config-wlc-policy-profile-ap)# description testpolicyprofile-descr
```

## rate-limit

Данная команда позволяет задать ограничение скорости для различных видов входящего и исходящего трафика.

Использование отрицательной формы команды (no) удаляет из профиля ограничение скорости на определенный вид трафика. При отсутствии параметров удаляются все установленные ограничения.

### Синтаксис

```
rate-limit { broadcast | multicast | station | vap } { input | output } { kbps | pps }
<TRAFFIC-RATE-VALUE>
no rate-limit [ broadcast | multicast | station | vap ] [ input | output ]
```

### Параметры

**broadcast** – ограничение скорости для широковещательного трафика. Не поддерживается на точках доступа WEP-2ac, WEP-2ac Smart, WOP-2ac, WOP-2ac:rev.B, WOP-2ac:rev.C и WEP-3ax;

**multicast** – ограничение скорости для многоадресного трафика. Не поддерживается на точках доступа WEP-2ac, WEP-2ac Smart, WOP-2ac, WOP-2ac:rev.B, WOP-2ac:rev.C и WEP-3ax;

**station** – ограничение скорости для каждого беспроводного клиента одной виртуальной точки доступа (VAP), на которой установлен SSID с данным профилем. Не поддерживается на точках доступа WEP-2ac, WEP-2ac Smart, WOP-2ac, WOP-2ac:rev.B, WOP-2ac:rev.C;

**vap** – ограничение скорости для всех беспроводных клиентов (в сумме) одной виртуальной точки доступа (VAP), на которой установлен SSID с данным профилем. Не поддерживается на точках доступа WEP-2ac, WEP-2ac Smart, WOP-2ac, WOP-2ac:rev.B, WOP-2ac:rev.C;

**input** – ограничение скорости будет установлено на трафик, идущий к беспроводному клиенту;

**output** – ограничение скорости будет установлено на трафик, идущий от беспроводного клиента;

**kbps** – определяет единицы измерения заданного ограничения скорости (Кбит/с);

**pps** – определяет единицы измерения заданного ограничения скорости (пакет в секунду);

**<TRAFFIC-RATE-VALUE>** – ограничение скорости, принимает значения от 0 до 2000000000. Единицы измерения определяются предыдущим параметром.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-POLICY-PROFILE-AP

```
wlc(config-wlc-policy-profile-ap)# rate-limit station input kbps 15000
```

## Настройка сервис-активатора

### service-activator

Данная команда позволяет перейти в раздел редактирования сервис-активатора.

Использование отрицательной формы команды (no) устанавливает настройки по умолчанию.

### Синтаксис

```
service-activator
```

### Параметры

Отсутствуют.

### Значение по умолчанию

Отсутствует

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC

### Пример

```
wlc(config-wlc)# service-activator
```

### aps join auto

Данная команда позволяет точкам доступа проходить регистрацию автоматически.

Использование отрицательной формы команды (no) позволяет регистрировать точки доступа вручную.

### Синтаксис

```
[no] aps join auto
```

### Параметры

Отсутствуют.

### Значение по умолчанию

Ручной режим регистрации.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-SERVICE-ACTIVATOR

### Пример

```
wlc(config-wlc-service-activator)# aps join auto
```

**password private-crt-key**

Данная команда задает пароль для сертификата точек доступа. Параметр необязательный. Использование отрицательной формы команды (no) удаляет пароль.

**Синтаксис**

```
password private-crt-key ascii-text <CLEAR-TEXT>
no password password private-crt-key
```

**Параметры**

<CLEAR-TEXT> – пароль, задается строкой [4-64] символов.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-SERVICE-ACTIVATOR

**Пример**

```
wlc(config-wlc-service-activator)# password private-crt-key ascii-text password
```

**port**

Данная команда указывает номер порта, на котором работает сервис. Сервис занимает два порта: указанный и следующий за ним.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
port <NUMBER>
no port
```

**Параметры**

<NUMBER> – номер порта, принимает значения от 1024 до 65535.

**Значение по умолчанию**

8043

**Необходимый уровень привилегий**

10

## Командный режим

CONFIG-WLC-SERVICE-ACTIVATOR

### Пример

```
wlc(config-wlc-service-activator)# port 8043
```

## Настройка SSID

### ssid-profile

Данная команда создает/назначает профиль SSID.

Использование отрицательной формы команды (no) удаляет SSID.

### Синтаксис

```
ssid-profile <NAME>  
no ssid-profile { <NAME> | all }
```

### Параметры

<NAME> – название профиля SSID, задается строкой до 235 символов;

all – команда удаляет все добавленные профили SSID.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC

CONFIG-WLC-AP-LOCATION

CONFIG-WLC-SSID-PROFILE

### Пример

```
wlc(config-wlc-ap-location)# ssid test_ssid
```

## 802.11k

Данная команда активирует работу по стандартам 802.11k/v.

Использование отрицательной формы команды (no) отключает работу по стандартам 802.11k/v.

**Синтаксис**

[no] 802.11kv

**Параметры**

Отсутствуют.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-SSID-PROFILE

**Пример**

```
wlc(config-wlc-ssid-profile)# 802.11kv
```

**802.11r**

Данная команда активирует работу по стандарту 802.11k.

Использование отрицательной формы команды (no) включает/отключает работу по стандарту 802.11r.

**Синтаксис**

[no] 802.11r

**Параметры**

Отсутствуют.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-SSID-PROFILE

**Пример**

```
wlc(config-wlc-ssid-profile)# 802.11r
```

## band

Данная команда назначает диапазон частот, в котором пользователи будут подключаться к беспроводной точке доступа.

Использование отрицательной формы команды (no) удаляет заданное значение диапазона частот.

### Синтаксис

```
band { 2g | 5g }  
no band
```

### Параметры

Отсутствуют.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-SSID-PROFILE

### Пример

```
wlc(config-wlc-ssid-radio)# band 2g
```

## band-steer-mode

Данная команда активирует на точках доступа функцию приоритетного подключения двухдиапазонных беспроводных клиентов к сети в 5 ГГц. Для работы функции необходимо, чтобы в ssid-profile были включены оба диапазона: band 5g и band 2g.

Использование отрицательной формы команды (no) деактивирует функцию.

### Синтаксис

```
[no] band-steer-mode
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

10



## Командный режим

CONFIG-WLC-SSID-PROFILE

### Пример

```
wlc(config-wlc-ssid-profile)# band-steer-mode
```

### check-signal-enable

Данная команда включает периодическую проверку сигнала.

Использование отрицательной формы команды (no) отменяет периодическую проверку сигнала.

### Синтаксис

```
[no] check-signal-enable
```

### Параметры

Отсутствуют.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-SSID-PROFILE

### Пример

```
wlc(config-wlc-ssid-profile)# check-signal-enable
```

### check-signal-timeout

Данная команда устанавливает время, через которое будет производиться периодическая проверка сигнала.

Использование отрицательной формы команды (no) отменяет периодическую проверку сигнала.

### Синтаксис

```
[no] check-signal-timeout <TIME>
```

### Параметры

<TIME> – промежуток времени (в секундах), через которое проводится периодическая проверка сигнала. Принимает значения в диапазоне [1...300].

**Значение по умолчанию**

10

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-SSID-PROFILE

**Пример**

```
wlc(config-wlc-ssid)# check-signal-timeout 10
```

**description**

Данная команда добавляет описание для блоков конфигурации.

Использование отрицательной формы команды (no) удаляет описание.

**Синтаксис**

```
[no] description <DESCRIPTION>
```

**Параметры**

<DESCRIPTION> – произвольное описание задается строкой до 255 символов.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-AIRTUNE-PROFILE

CONFIG-WLC-AP

CONFIG-WLC-AP-PROFILE

CONFIG-WLC-IP-POOL

CONFIG-WLC-AP-LOCATION

CONFIG-WLC-PORTAL-PROFILE

CONFIG-WLC-RADIUS-PROFILE

CONFIG-WLC-SSID-PROFILE

CONFIG-WLC-WIDS-PROFILE

## Пример

```
wlc(config-wlc-ip-pool)# description default_test_pool
```

## enable

Данная команда предназначена для активации функций и сервисов контроллера или точек доступа. Использование отрицательной формы команды (no) отключает функции.

## Синтаксис

```
[no] enable
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Отключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC

CONFIG-WLC-SSID-PROFILE

CONFIG-AIRTUNE

## Пример

```
wlc(config-wlc)# enable
```

## general-vlan-id

Данная команда устанавливает идентификатор General VLAN.

Использование отрицательной формы команды (no) удаляет заданный идентификатор.

## Синтаксис

```
general-vlan-id <ID>  
no general-vlan-id
```

## Параметры

<ID> – идентификатор General VLAN, принимает значения в диапазоне [0-4094] ("0" предназначен для удаления параметра).

## Значение по умолчанию

0

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-SSID-PROFILE

**Пример**

```
wlc(config-wlc-ssid-profile)# general-vlan-id 1
```

**general-vlan-mode**

Данная команда активирует использование General VLAN.

Использование отрицательной формы команды (no) деактивирует использование General VLAN.

**Синтаксис**

```
[no] general-vlan-mode
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-SSID-PROFILE

**Пример**

```
wlc(config-wlc-ssid-profile)# general-vlan-mode
```

**hidden**

Данная команда скрывает SSID.

Использование отрицательной формы команды (no) деактивирует функционал.

**Синтаксис**

```
[no] hidden
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-SSID-PROFILE

**Пример**

```
wlc(config-wlc-ssid-profile)# hidden
```

**inactivity-timeout**

Данная команда устанавливает время бездействия, после которого пользователь будет считаться неактивным.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
inactivity-timeout <TIMEOUT>  
no inactivity-timeout
```

**Параметры**

<TIMEOUT> – временной промежуток, принимает значения в диапазоне [0-86400].

**Значение по умолчанию**

120

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-SSID-PROFILE

**Пример**

```
wlc(config-wlc-ssid-profile)# inactivity-timeout 120
```

**key-wpa**

Данная команда устанавливает ключ/пароль для подключения к виртуальной точке доступа. Пароль может быть задан как в открытом виде, так и в виде хеш sha512.

Использование отрицательной формы команды (no) удаляет заданный ключ.

### Синтаксис

```
key-wpa { ascii-text <CLEAR-TEXT> | encrypted <HASH_SHA512> }  
no key-wpa
```

### Параметры

<CLEAR-TEXT> – ключ, задается строкой [8-63] символа;

<HASH\_SHA512> – хеш ключа по алгоритму sha512, задается строкой [16-126] символов.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-SSID-PROFILE

### Пример

```
wlc(config-wlc-ssid-profile)# key-wpa password
```

### local-switching

Данная команда активирует функцию local-switching.

Использование отрицательной формы команды (no) отключает функцию.

### Синтаксис

```
[no] local-switching
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-SSID-PROFILE

### Пример

```
wlc(config-wlc-ssid-profile)# local-switching
```

**mac-auth mode local**

Данная команда активирует функцию MAC-авторизации клиентов по локальным спискам. Использование отрицательной формы команды (no) отключает функцию.

**Синтаксис**

```
mac-auth mode local policy { <permit> | <deny> } { <object-group> | <any> }
no mac-auth
```

**Параметры**

<permit> – разрешить авторизацию;

<deny> – запретить авторизацию;

<object-group> – название object-group, задается строкой до 31 символа, должно совпадать с существующей object-group.

При указании значения «any» функция будет срабатывать для любого устройства.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-SSID-PROFILE

**Пример**

```
wlc(config-wlc-ssid-profile)# mac-auth mode local policy deny any
```

**mac-auth mode radius**

Данная команда активирует функцию MAC-авторизации клиентов по записям на RADIUS-сервере. Использование отрицательной формы команды (no) отключает функцию.

**Синтаксис**

```
mac-auth mode radius policy { <permit> | <deny> }
no mac-auth
```

**Параметры**

<permit> – разрешить авторизацию для устройств, указанных в RADIUS;

<deny> – запретить авторизацию для устройств, указанных в RADIUS.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-SSID-PROFILE

**Пример**

```
wlc(config-wlc-ssid-profile)# mac-auth mode radius permit
```

**minimal-signal**

Данная команда устанавливает пороговое значение RSSI, при достижении которого точка доступа будет отключать клиента от виртуальной точки доступа.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
minimal-signal <RSSI>  
no minimal-signal
```

**Параметры**

<RSSI> – пороговое значение RSSI, принимает значения в диапазоне [-100.. 0].

**Значение по умолчанию**

-100

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-SSID-PROFILE

**Пример**

```
wlc(config-wlc-ssid-profile)# minimal-signal -100
```

**pmksa-caching**

Данная команда включает кэширование информации о подключении клиента. При включении данной функции точка доступа запоминает клиентское устройство на 12 часов и не требует повторной аутентификации при подключении в течение этого времени. Включение данной функции сокращает время роуминга при возвращении клиента на точку в режиме WPA Enterprise.

Использование отрицательной формы команды (no) деактивирует функцию.



### Синтаксис

[no] pmksa-caching

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-SSID-PROFILE

### Пример

```
wlc(config-wlc-ssid-profile)# pmksa-caching
```

### policy-profile ap

Данная команда назначает профиль ограничения скорости.

Использование отрицательной формы команды (no) удаляет профиль ограничения скорости из SSID профиля. При отсутствии параметров удаляются все установленные профили.

### Синтаксис

```
policy-profile ap band { 2g | 5g | all } <POLICY-NAME>  
no policy-profile ap [ 2g | 5g | all ]
```

### Параметры

<POLICY-NAME> – название профиля, задается строкой до 235 символов.

2g – ограничения профиля будут применяться только для трафика в диапазоне 2,4 ГГц.

5g – ограничения профиля будут применяться только для трафика в диапазоне 5 ГГц.

all – ограничения профиля будут применяться для трафика в диапазонах 2,4 и 5 ГГц.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-SSID-PROFILE

**Пример**

```
wlc(config-wlc-ssid-profile)# policy-profile ap band all testpolicyprofile
```

**portal-enabled**

Данная команда активирует порталную авторизацию.

Использование отрицательной формы команды (no) деактивирует функцию.

**Синтаксис**

```
[no] portal-enabled
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-SSID-PROFILE

**Пример**

```
wlc(config-wlc-ssid-profile)# portal-enabled
```

**portal-profile**

Данная команда назначает профиль портала.

Использование отрицательной формы команды (no) удаляет настройку профиля.

**Синтаксис**

```
portal-profile <PORTAL-NAME>
no portal-profile [ <PORTAL-NAME> | all ]
```

**Параметры**

<PORTAL-NAME> – название профиля портала, задается строкой до 235 символов;

all – команда удаляет все созданные профили порталов.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC

CONFIG-WLC-SSID-PROFILE

**Пример**

```
wlc(config-wlc)# portal-profile test
```

**priority-by-dscp**

Данная команда активирует на точках доступа анализ приоритета из поля DSCP заголовка IP-пакета для распределения трафика, передающегося в радиointерфейс, по очередям WMM. Если значение DSCP в тегированных кадрах равно 0, то анализироваться будет приоритет из поля CoS (Class of Service).

Использование отрицательной формы команды (no) отключает анализ приоритета по полю DSCP. В таком случае будет анализироваться приоритет из поля CoS (802.1p) тегированных кадров.

**Синтаксис**

```
[no] priority-by-dscp
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-SSID-PROFILE

**Пример**

```
wlc(config-wlc-ssid-profile)# priority-by-dscp
```

**radius-profile**

Данная команда назначает/настраивает профиль RADIUS-сервера.

Использование отрицательной формы команды (no) отменяет назначение/настройки профиля RADIUS-сервера.

**Синтаксис**

```
radius-profile <RADIUS-ID>
no radius-profile { <RADIUS-ID> | all }
```

**Параметры**

<RADIUS-ID> – идентификатор RADIUS-сервера, задается строкой до 235 символов;  
all – команда удаляет все созданные RADIUS-сервера.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC

CONFIG-WLC-PORTAL-PROFILE

CONFIG-WLC-SSID-PROFILE

**Пример**

```
wlc(config-wlc)# radius-profile test
```

**roaming-signal**

Данная команда устанавливает уровень RSSI, при достижении которого будет срабатывать роуминг.  
Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
roaming-signal <RSSI>
no roaming-signal
```

**Параметры**

<RSSI> – уровень RSSI, принимает значения от -100 до -1.

**Значение по умолчанию**

-100

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-SSID-PROFILE

**Пример**

```
wlc(config-wlc-ssid-profile)# roaming-signal -100
```

**security-mode**

Данная команда устанавливает режим безопасности на виртуальной точке доступа.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
security-mode <MODE>
no security-mode
```

**Параметры**

<MODE> – режим безопасности, доступные значения:

- OWE
- WPA;
- WPA2;
- WPA2\_1X;
- WPA2\_WPA3;
- WPA2\_WPA3\_1X;
- WPA3;
- WPA3\_1X;
- WPA\_1X;
- WPA\_WPA2;
- WPA\_WPA2\_1X;
- off.

Режимы безопасности WPA3 и OWE поддерживаются на точках доступа моделей WEP-3ax, WEP-3L, WEP-30L, WEP-30L-Z, WOP-30L, WOP-30LS, WOP-30LI

При выборе смешанного режима безопасности (например, WPA2\_WPA3) WPA3 будет применен только для тех точек доступа, которые его поддерживают, для остальных будет применен второй режим (WPA2).

**Значение по умолчанию**

off

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-SSID-PROFILE

**Пример**

```
wlc(config-wlc-ssid-profile)# security-mode WPA
```

## ssid

Данная команда задает имя SSID, которое будет вещаться пользователям.

Использование отрицательной формы команды (no) удаляет название SSID.

### Синтаксис

```
ssid <NAME>  
no ssid
```

### Параметры

<NAME> – название SSID, задается строкой до 32 символов. Названия, содержащие пробел, необходимо заключать в кавычки.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-SSID-PROFILE

### Пример

```
wlc(config-wlc-ssid-profile)# ssid 'test ssid'
```

## sta-limit

Данная команда задает максимальное число пользователей, которые могут подключиться к виртуальной точке доступа.

Использование отрицательной формы команды (no) назначает значение по умолчанию.

### Синтаксис

```
sta-limit <LIMIT>  
no sta-limit
```

### Параметры

<LIMIT> – максимальное число пользователей, принимает значения в диапазоне [0-64] ("0" служит для отключения ограничения).

### Значение по умолчанию

0

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-SSID-PROFILE

### Пример

```
wlc(config-wlc-ssid-profile)# sta-limit 64
```

### station-isolation

Данная команда активирует изоляцию трафика между клиентами в пределах одной виртуальной точки доступа.

Использование отрицательной формы команды (no) отключает изоляцию.

### Синтаксис

```
[no] station-isolation
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-SSID-PROFILE

### Пример

```
wlc(config-wlc-ssid-profile)# station-isolation
```

### vlan-id

Данная команда назначает vlan на виртуальную точку доступа.

Использование отрицательной формы команды (no) применяет значение по умолчанию.

### Синтаксис

```
vlan-id <ID>  
no vlan-id
```

### Параметры

<ID> – идентификатор vlan, принимает значения в диапазоне [0-4094].

**Значение по умолчанию**

0

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-SSID-PROFILE

**Пример**

```
wlc(config-wlc-ssid-profile)# vlan-id 1111
```

**vlan-priority**

Данная команда назначает приоритет на клиентский трафик.

Использование отрицательной формы команды (no) применяет значение по умолчанию.

**Синтаксис**

```
vlan-priority <PRIORITY>  
no vlan-priority
```

**Параметры**

<PRIORITY> – приоритет, доступные значения:

- auto;
- значения из диапазона [0.. 7].

**Значение по умолчанию**

auto

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-SSID-PROFILE

**Пример**

```
wlc(config-wlc-ssid-profile)# vlan-priority auto
```

**vlan-trunk**

Данная команда активирует функцию Vlan Trunk.

Использование отрицательной формы команды (no) деактивирует функцию.



**Синтаксис**

[no] vlan-trunk

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-SSID-PROFILE

**Пример**

```
wlc(config-wlc-ssid-profile)# vlan-trunk
```

**Настройки сервиса WIDS/WIPS****wids**

Данная команда позволяет перейти в общие настройки сервиса WIPS/WIDS. Это внутренний сервис точки доступа по обнаружению и предотвращению вторжений в беспроводную сеть. Функционал активируется специальной лицензией.

Использование отрицательной формы команды (no) устанавливает значения сконфигурированных настроек WIDS по умолчанию.

**Синтаксис**

[no] wids

**Параметры**

Отсутствуют.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC

## Пример

```
wlc(config-wlc)# wids
wlc(config-wlc-wids)#
```

## enable

Данная команда предназначена для активации функций и сервисов контроллера или точек доступа. Использование отрицательной формы команды (no) отключает функции.

## Синтаксис

```
[no] enable
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Отключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC

CONFIG-WLC-SSID-PROFILE

CONFIG-WLC-AIRTUNE

CONFIG-WLC-WIDS

## Пример

```
wlc(config-wlc)# enable
```

## shared-key

Данная команда задает общий ключ, используемый для отслеживания доверенных точек доступа в радиоэфире. Для активации сервиса установка ключа является обязательной. Ключ может быть задан как в открытом виде, так и в виде хеш sha512.

Использование отрицательной формы команды (no) удаляет ключ.

## Синтаксис

```
shared-key ascii-text { <CLEAR-TEXT> | encrypted <HASH_SHA512> }
no shared-key
```

## Параметры

<CLEAR-TEXT> – ключ, задается строкой [10-32] символов;

<HASH\_SHA512> – хеш ключа по алгоритму sha512, задается строкой [20-64] символов.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-WIDS

### Пример

```
wlc(config-wlc-wids)# shared-key ascii-text 0123456789
```

### wids-profile

Данная команда назначает профиль WIDS в общих настройках сервиса.

Использование отрицательной формы команды (no) устанавливает профиль WIDS по умолчанию.

### Синтаксис

```
wids-profile <NAME>  
no wids-profile
```

### Параметры

<NAME> – название профиля WIDS, задается строкой до 235 символов.

### Значение по умолчанию

default-wids

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-WIDS

### Пример

```
wlc(config-wlc-wids)# wids-profile test
```

### white-list

Данная команда назначает списки MAC-адресов "доверенных" точек доступа, которые сформированы в [object-group mac](#). "Доверенными" считаются точки доступа, которые установлены и управляются оператором;

Использование отрицательной формы команды (no) отменяет использование белого списка.

Списки MAC-адресов могут быть заданы на трех уровнях: в общих настройках WIDS сервиса, в локации и в индивидуальном профиле точки доступа. Приоритет применения списков следующий: настройки индивидуального профиля, настройки локации, общие настройки сервиса. В случае задания на одном уровне только списка одного вида (белого или черного), противоположный список будет использован со следующего уровня. Например, если в индивидуальном профиле точки доступа задан только белый список, то черный будет взят из настроек локации. Если он не задан в локации, то будет взят из общих настроек сервиса. Если требуется задать пустой список, то необходимо создать пустую группу MAC-адресов и использовать ее на нужном уровне.

### Синтаксис

```
white-list <OBJECT-GROUP-MAC>
no white-list
```

### Параметры

<OBJECT-GROUP-MAC> – название существующей группы MAC-адресов, задается строкой до 31 символа.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-WIDS

### Пример

```
wlc(config-wlc-wids)# white-list test
```

### black-list

Данная команда назначает списки MAC-адресов "вражеских" точек доступа, которые сформированы в [object-group mac](#). "Вражескими" считаются точки, которые несут угрозу для остальных точек доступа в сети, как правило, это известные ТД, которые были обнаружены имитирующими MAC-адрес или SSID исходной ТД.

Использование отрицательной формы команды (no) отменяет использование черного списка.

Списки MAC-адресов могут быть заданы на трех уровнях: в общих настройках WIDS сервиса, в локации и в индивидуальном профиле точки доступа. Приоритет применения списков следующий: настройки индивидуального профиля, настройки локации, общие настройки сервиса. В случае задания на одном уровне только списка одного вида (белого или черного), противоположный список будет использован со следующего уровня. Например, если в индивидуальном профиле точки доступа задан только черный список, то белый будет взят из настроек локации. Если он не задан в локации, то будет взят из общих настроек сервиса. Если требуется задать пустой список, то необходимо создать пустую группу MAC-адресов и использовать ее на нужном уровне.

### Синтаксис

```
black-list <OBJECT-GROUP-MAC>
no black-list
```

## Параметры

<OBJECT-GROUP-MAC> – название существующей группы MAC-адресов, задается строкой до 31 символа.

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-WIDS

## Пример

```
wlc(config-wlc-wids)# black-list test
```

## wids-profile

Данная команда создает профиль настроек WIDS сервиса.

Использование отрицательной формы команды (no) удаляет один выбранный профиль WIDS или все существующие.

## Синтаксис

```
wids-profile <NAME>  
no wids-profile { <NAME> | all }
```

## Параметры

<NAME> – название профиля WIDS, задается строкой до 235 символов.

all – команда удаляет все созданные профили WIDS.

## Значение по умолчанию

default-wids

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC

## Пример

```
wlc(config-wlc)# wids-profile test
```

## **bruteforce-detection**

Данная команда осуществляет переход в режим настройки атаки WIDS типа Bruteforce.

Использование отрицательной формы команды (no) возвращает значения по умолчанию для всех сконфигурированных настроек атаки Bruteforce.

### **Синтаксис**

```
[no] bruteforce-detection
```

### **Параметры**

Команда не содержит параметров.

### **Необходимый уровень привилегий**

10

### **Командный режим**

CONFIG-WLC-WIDS-PROFILE

### **Пример**

```
wlc(config-wlc-wids-profile)# bruteforce-detection
```

## **enable**

Данная команда предназначена для активации функций и сервисов контроллера или точек доступа.

Использование отрицательной формы команды (no) отключает функции.

### **Синтаксис**

```
[no] enable
```

### **Параметры**

Команда не содержит параметров.

### **Значение по умолчанию**

Отключено.

### **Необходимый уровень привилегий**

10

### **Командный режим**

CONFIG-WLC

CONFIG-WLC-SSID-PROFILE

CONFIG-WLC-AIRTUNE

CONFIG-WLC-WIDS

CONFIG-WLC-WIDS-PROFILE-BF-DETECTION

## CONFIG-WLC-WIDS-PROFILE-DOS-DETECTION

### Пример

```
wlc(config-wlc-wids-profile-bf-detection)# enable
```

### threshold

Данная команда задает пороговое значение количества неуспешных авторизаций для блокировки. Использование отрицательной формы команды (no) возвращает значение по умолчанию.

### Синтаксис

```
threshold <THRESHOLD>  
[no] threshold
```

### Параметры

<THRESHOLD> – пороговое значение количества неуспешных авторизаций для блокировки, возможные значения от 1 до 10000.

### Значение по умолчанию

25

### Необходимый уровень привилегий

10

### Командный режим

## CONFIG-WLC-WIDS-PROFILE-BF-DETECTION

### Пример

```
wlc(config-wlc-wids-profile-bf-detection)# threshold 50
```

### interval

Данная команда задает интервал подсчёта неудачных попыток авторизаций. В течение указанного интервала считается количество неуспешных авторизаций пользователей на SSID с шифрованием (Personal и Enterprise) на ТД. Если порог был превышен, на контроллер отправляется сообщение об обнаружении атаки "перебор паролей".

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

### Синтаксис

```
interval <TIME>  
[no] interval
```

### Параметры

<TIME> – интервал подсчёта неудачных попыток авторизаций, возможные значения от 0 до 86400 секунд. При значении параметра равного 0, детектирование атаки "перебор паролей" будет отключено.

**Значение по умолчанию**

5

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-WIDS-PROFILE-BF-DETECTION

**Пример**

```
wlc(config-wlc-wids-profile-bf-detection)# interval 50
```

**mac-ban enable**

Данная команда включает функцию блокировки клиентских устройств, замеченных за атакой "перебор паролей". MAC-адреса клиентских устройств будут добавлены в "черный" список и будут игнорироваться точкой доступа в течение периода времени, заданного в параметре "Timeout", в результате чего, клиент не сможет подключиться к сети.

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

**Синтаксис**

```
mac-ban enable  
[no] mac-ban enable
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-WIDS-PROFILE-BF-DETECTION

**Пример**

```
wlc(config-wlc-wids-profile-bf-detection)# mac-ban enable
```

**mac-ban timeout**

Данная команда задает продолжительность блокировки клиентских устройств, замеченных за атакой "перебор паролей".

Использование отрицательной формы команды (no) возвращает значение по умолчанию.



## Синтаксис

```
mac-ban timeout <TIME>  
[no] mac-ban timeout
```

## Параметры

<TIME> – время хранения MAC-адреса клиентского устройства в "черном" списке, возможные значения от 0 до 86400 секунд.

## Значение по умолчанию

1800

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-WIDS-PROFILE-BF-DETECTION

## Пример

```
wlc(config-wlc-wids-profile-bf-detection)# mac-ban timeout 60
```

## description

Данная команда добавляет описание для блоков конфигурации.  
Использование отрицательной формы команды (no) удаляет описание.

## Синтаксис

```
description <DESCRIPTION>  
[no] description
```

## Параметры

<DESCRIPTION> – произвольное описание, задается строкой до 255 символов.

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-AIRTUNE-PROFILE  
CONFIG-WLC-AP  
CONFIG-WLC-AP-PROFILE  
CONFIG-WLC-IP-POOL

CONFIG-WLC-AP-LOCATION  
CONFIG-WLC-PORTAL-PROFILE  
CONFIG-WLC-RADIUS-PROFILE  
CONFIG-WLC-SSID-PROFILE  
CONFIG-WLC-RADIO-2G-PROFILE  
CONFIG-WLC-RADIO-5G-PROFILE  
CONFIG-WLC-WIDS-PROFILE

### Пример

```
wlc(config-wlc-wids-profile)# description test_wids_profile
```

### dos-detection

Данная команда осуществляет переход в режим настройки DoS-атаки WIDS.

Использование отрицательной формы команды (no) возвращает значения по умолчанию для всех сконфигурированных настроек DoS-атаки.

### Синтаксис

```
[no] dos-detection
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-WIDS-PROFILE

### Пример

```
wlc(config-wlc-wids-profile)# dos-detection
```

### enable

Данная команда предназначена для активации функций и сервисов контроллера или точек доступа.

Использование отрицательной формы команды (no) отключает функции.

### Синтаксис

```
[no] enable
```

### Параметры

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC

CONFIG-WLC-SSID-PROFILE

CONFIG-WLC-AIRTUNE

CONFIG-WLC-WIDS

CONFIG-WLC-WIDS-PROFILE-BF-DETECTION

CONFIG-WLC-WIDS-PROFILE-DOS-DETECTION

```
wlc(config-wlc-wids-profile-dos-detection)# enable
```

**interval**

Данная команда задает интервал подсчёта пакетов в радиозфире. Если за это время порог, установленный для какого-то типа пакетов, был превышен, формируется сообщение об обнаружении DoS-атаки.

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

**Синтаксис**

```
interval <TIME>
[no] interval
```

**Параметры**

<TIME> – интервал подсчёта пакетов в радиозфире, возможные значения от 0 до 86400 секунд.

**Значение по умолчанию**

1

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-WIDS-PROFILE-DOS-DETECTION

**Пример**

```
wlc(config-wlc-wids-profile-dos-detection)# interval 5
```

## trap-send-period

Данная команда задает период отправки сообщений об обнаружении DoS-атак.

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

### Синтаксис

```
trap-send-period <TIME>  
[no] trap-send-period
```

### Параметры

<TIME> – период отправки сообщений об обнаружении DoS-атаки, возможные значения от 0 до 604800 секунд.

### Значение по умолчанию

20

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-WIDS-PROFILE-DOS-DETECTION

### Пример

```
wlc(config-wlc-wids-profile-dos-detection)# trap-send-period 3600
```

## threshold assoc

Данная команда задает пороговое значение количества пакетов типа 'Request for association'. Если порог был превышен за заданный интервал времени, формируется сообщение об обнаружении DoS-атаки.

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

### Синтаксис

```
threshold assoc <THRESHOLD>  
[no] threshold assoc
```

### Параметры

<THRESHOLD> – пороговое значение количества пакетов типа 'Request for association', возможные значения от 0 до 86400 секунд.

### Значение по умолчанию

500

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-WIDS-PROFILE-DOS-DETECTION

### Пример

```
wlc(config-wlc-wids-profile-dos-detection)# threshold assoc 100
```

### threshold auth

Данная команда задает пороговое значение количества пакетов типа 'Authentication'. Если порог был превышен за заданный интервал времени, формируется сообщение об обнаружении DoS-атаки.

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

### Синтаксис

```
threshold auth <THRESHOLD>  
[no] threshold auth
```

### Параметры

<THRESHOLD> – пороговое значение количества пакетов типа 'Authentication', возможные значения от 0 до 86400 секунд.

### Значение по умолчанию

500

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-WIDS-PROFILE-DOS-DETECTION

### Пример

```
wlc(config-wlc-wids-profile-dos-detection)# threshold auth 100
```

### threshold beacon

Данная команда задает пороговое значение количества пакетов типа 'Beacon'. Если порог был превышен за заданный интервал времени, формируется сообщение об обнаружении DoS-атаки.

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

### Синтаксис

```
threshold beacon <THRESHOLD>  
[no] threshold beacon
```

## Параметры

<THRESHOLD> – пороговое значение количества пакетов типа 'Beacon', возможные значения от 0 до 86400 секунд.

## Значение по умолчанию

500

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-WIDS-PROFILE-DOS-DETECTION

## Пример

```
wlc(config-wlc-wids-profile-dos-detection)# threshold beacon 100
```

## threshold blockack

Данная команда задает пороговое значение количества пакетов типа 'Block Ack'. Если порог был превышен за заданный интервал времени, формируется сообщение об обнаружении DoS-атаки.

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

## Синтаксис

```
threshold blockack <THRESHOLD>  
[no] threshold blockack
```

## Параметры

<THRESHOLD> – пороговое значение количества пакетов типа 'Block Ack', возможные значения от 0 до 86400 секунд.

## Значение по умолчанию

500

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-WIDS-PROFILE-DOS-DETECTION

## Пример

```
wlc(config-wlc-wids-profile-dos-detection)# threshold blockack 100
```

**threshold blockack-req**

Данная команда задает пороговое значение количества пакетов типа 'Block Ack request'. Если порог был превышен за заданный интервал времени, формируется сообщение об обнаружении DoS-атаки.

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

**Синтаксис**

```
threshold blockack-req <THRESHOLD>
[no] threshold blockack-req
```

**Параметры**

<THRESHOLD> – пороговое значение количества пакетов типа 'Block Ack request', возможные значения от 0 до 86400 секунд.

**Значение по умолчанию**

500

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-WIDS-PROFILE-DOS-DETECTION

**Пример**

```
wlc(config-wlc-wids-profile-dos-detection)# threshold blockack-req 100
```

**threshold cts**

Данная команда задает пороговое значение количества пакетов типа 'CTS'. Если порог был превышен за заданный интервал времени, формируется сообщение об обнаружении DoS-атаки.

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

**Синтаксис**

```
threshold cts <THRESHOLD>
[no] threshold cts
```

**Параметры**

<THRESHOLD> – пороговое значение количества пакетов типа 'CTS', возможные значения от 0 до 86400 секунд.

**Значение по умолчанию**

500

**Необходимый уровень привилегий**

10

## Командный режим

CONFIG-WLC-WIDS-PROFILE-DOS-DETECTION

### Пример

```
wlc(config-wlc-wids-profile-dos-detection)# threshold cts 100
```

### threshold deauth

Данная команда задает пороговое значение количества пакетов типа 'Deauthentication'. Если порог был превышен за заданный интервал времени, формируется сообщение об обнаружении DoS-атаки.

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

### Синтаксис

```
threshold deauth <THRESHOLD>  
[no] threshold deauth
```

### Параметры

<THRESHOLD> – пороговое значение количества пакетов типа 'Deauthentication', возможные значения от 0 до 86400 секунд.

### Значение по умолчанию

500

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-WIDS-PROFILE-DOS-DETECTION

### Пример

```
wlc(config-wlc-wids-profile-dos-detection)# threshold deauth 100
```

### threshold disassoc

Данная команда задает пороговое значение количества пакетов типа 'Disassociation request'. Если порог был превышен за заданный интервал времени, формируется сообщение об обнаружении DoS-атаки.

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

### Синтаксис

```
threshold disassoc <THRESHOLD>  
[no] threshold disassoc
```



## Параметры

<THRESHOLD> – пороговое значение количества пакетов типа 'Disassociation request', возможные значения от 0 до 86400 секунд.

## Значение по умолчанию

500

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-WIDS-PROFILE-DOS-DETECTION

## Пример

```
wlc(config-wlc-wids-profile-dos-detection)# threshold disassoc 100
```

## threshold leap

Данная команда задает пороговое значение для изменения количества пакетов относительно предыдущего периода времени. Параметр показывает, насколько должно измениться количество пакетов по сравнению с предыдущим периодом, чтобы было отправлено сообщение о DoS-атаке на контроллер.

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

## Синтаксис

```
threshold leap <THRESHOLD>  
[no] threshold leap
```

## Параметры

<THRESHOLD> – пороговое значение изменения количества пакетов относительно предыдущего периода времени, возможные значения от 0 до 86400 секунд.

## Значение по умолчанию

250

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-WIDS-PROFILE-DOS-DETECTION

## Пример

```
wlc(config-wlc-wids-profile-dos-detection)# threshold leap 100
```

**threshold probe**

Данная команда задает пороговое значение количества пакетов типа 'Probe request' и 'Probe response'. Если порог был превышен за заданный интервал времени, формируется сообщение об обнаружении DoS-атаки.

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

**Синтаксис**

```
threshold probe <THRESHOLD>
[no] threshold probe
```

**Параметры**

<THRESHOLD> – пороговое значение количества пакетов типа 'Probe request' и 'Probe response', возможные значения от 0 до 86400 секунд.

**Значение по умолчанию**

500

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-WIDS-PROFILE-DOS-DETECTION

**Пример**

```
wlc(config-wlc-wids-profile-dos-detection)# threshold probe 100
```

**threshold ps-poll**

Данная команда задает пороговое значение количества пакетов типа 'PS poll'. Если порог был превышен за заданный интервал времени, формируется сообщение об обнаружении DoS-атаки.

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

**Синтаксис**

```
threshold ps-poll <THRESHOLD>
[no] threshold ps-poll
```

**Параметры**

<THRESHOLD> – пороговое значение количества пакетов типа 'PS poll', возможные значения от 0 до 86400 секунд.

**Значение по умолчанию**

500

**Необходимый уровень привилегий**

10

## Командный режим

CONFIG-WLC-WIDS-PROFILE-DOS-DETECTION

### Пример

```
wlc(config-wlc-wids-profile-dos-detection)# threshold ps-poll 100
```

### threshold reassoc

Данная команда задает пороговое значение количества пакетов типа 'Reassociation request'. Если порог был превышен за заданный интервал времени, формируется сообщение об обнаружении DoS-атаки.

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

### Синтаксис

```
threshold reassoc <THRESHOLD>  
[no] threshold reassoc
```

### Параметры

<THRESHOLD> – пороговое значение количества пакетов типа 'Reassociation request', возможные значения от 0 до 86400 секунд.

### Значение по умолчанию

500

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-WIDS-PROFILE-DOS-DETECTION

### Пример

```
wlc(config-wlc-wids-profile-dos-detection)# threshold reassoc 100
```

### threshold rts

Данная команда задает пороговое значение количества пакетов типа 'RTS'. Если порог был превышен за заданный интервал времени, формируется сообщение об обнаружении DoS-атаки.

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

### Синтаксис

```
threshold rts <THRESHOLD>  
[no] threshold rts
```

## Параметры

<THRESHOLD> – пороговое значение количества пакетов типа 'RTS' , возможные значения от 0 до 86400 секунд.

## Значение по умолчанию

500

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-WIDS-PROFILE-DOS-DETECTION

## Пример

```
wlc(config-wlc-wids-profile-dos-detection)# threshold rts 100
```

## scan

Данная команда осуществляет переход к настройкам сканирования WIDS.

Использование отрицательной формы команды (no) возвращает значения по умолчанию для всех сконфигурированных настроек сканирования.

## Синтаксис

[no] scan

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-WIDS-PROFILE

## Пример

```
wlc(config-wlc-wids-profile)# scan
```

## interface

Данная команда задает интерфейс сканирования WIDS.

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

**Синтаксис**

```
interface {all | wlan0 | wlan1}
[no] interface
```

**Параметры**

all – сканирование будут осуществлять оба радиointерфейса точки доступа: в диапазонах 2,4 ГГц и 5 ГГц;

wlan0 – сканирование будет осуществлять только радиointерфейс ТД в диапазоне 2,4 ГГц;

wlan1 – сканирование будет осуществлять только радиointерфейс ТД в диапазоне 5 ГГц.

**Значение по умолчанию**

all

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-WIDS-PROFILE-SCAN

**Пример**

```
wlc(config-wlc-wids-profile-scan)# interface wlan0
```

**mode**

Данная команда позволяет выбрать режим сканирования WIDS.

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

**Синтаксис**

```
mode {none | passive | sentry}
[no] mode
```

**Параметры**

none – сканирование выключено;

passive – включено пассивное сканирование. В этом режиме точка доступа через заданные промежутки времени будет кратковременно менять свой текущий канал (на котором идет работа с клиентами) на очередной канал из общего списка для обнаружения других ТД в эфире. Качество услуги, предоставляемой клиенту, в момент сканирования практически не деградирует;

sentry – включено активное сканирование. В этом режиме не предусмотрена работа точки доступа с клиентами. Точка доступа все время сканирует весь список каналов и максимально быстро обнаруживает угрозы.

**Значение по умолчанию**

none

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-WIDS-PROFILE-SCAN

**Пример**

```
wlc(config-wlc-wids-profile-scan)# mode passive
```

**passive interval**

Данная команда задает период пассивного сканирования WIDS.

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

**Синтаксис**

```
passive interval <TIME>  
[no] passive interval
```

**Параметры**

<TIME> – время в секундах между сканированием в пассивном режиме, возможные значения от 1 до 3600 секунд.

**Значение по умолчанию**

20

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-WIDS-PROFILE-SCAN

**Пример**

```
wlc(config-wlc-wids-profile-scan)# passive interval 30
```

**passive time**

Данная команда задает продолжительность сканирования на одном радиоканале в пассивном режиме.

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

**Синтаксис**

```
passive time <TIME>  
[no] passive time
```

## Параметры

<TIME> – длительность пассивного сканирования на одном канале в миллисекундах, возможные значения от 10 до 2000 мс.

## Значение по умолчанию

110

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-WIDS-PROFILE-SCAN

## Пример

```
wlc(config-wlc-wids-profile-scan)# passive time 30
```

## sentry time

Данная команда задает продолжительность сканирования на одном радиоканале в активном режиме. Использование отрицательной формы команды (no) возвращает значение по умолчанию.

## Синтаксис

```
sentry time <TIME>  
[no] sentry time
```

## Параметры

<TIME> – длительность активного сканирования на одном канале в миллисекундах, возможные значения от 100 до 2000 мс.

## Значение по умолчанию

200

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-WIDS-PROFILE-SCAN

## Пример

```
wlc(config-wlc-wids-profile-scan)# sentry time 300
```

**prevention-mode**

Данная команда включает режим подавления угроз (WIPS).

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

**Синтаксис**

```
prevention-mode {none | rogue | all}
[no] prevention-mode
```

**Параметры**

none – режим подавления угроз выключен;

rogue – сканирующая ТД детектирует MAC-адреса клиентов, которые подключены к "вражеским" ТД, и отправляет DeAuth пакеты от имени "вражеской" ТД клиенту.

all – в данном режиме форсированный DeAuth пакет отправляется не только клиентам, подключенным к "вражеским" ТД, а вообще всем, кто подключен к "недоверенным" ТД.

**Значение по умолчанию**

none

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-WIDS-PROFILE

**Пример**

```
wlc(config-wlc-wids-profile)# prevention-mode rogue
```

**attack-stats-trap-send-period**

Данная команда задает период отправки на контроллер сообщений, содержащих статистику срабатываний функционала подавления угроз (WIPS).

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

**Синтаксис**

```
attack-stats-trap-send-period <TIME>
[no] attack-stats-trap-send-period
```

**Параметры**

<TIME> – период отправки сообщений на контроллер в минутах, возможные значения от 1 до 1440 минут.

**Значение по умолчанию**

10



## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-WIDS-PROFILE

## Пример

```
wlc(config-wlc-wids-profile)# attack-stats-trap-send-period 60
```

## Настройка менеджера обновления ПО

### update-manager

Данная команда осуществляет переход в режим настройки менеджера обновления ПО точек доступа по расписанию.

Использование отрицательной формы команды (no) удаляет сконфигурированные настройки.

### Синтаксис

```
[no] update-manager
```

### Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC

## Пример

```
wlc(config-wlc)# update-manager
```

### allow-update-with-clients

Данная команда позволяет обновлять точки доступа, на которых в данный момент подключены клиенты.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
[no] allow-update-with-clients
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Отключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-UPDATE-MANAGER

## Пример

```
wlc(config-wlc-update-manager)# allow-update-with-clients
```

## end-time

Данная команда задает окончание временного интервала, в котором производится обновление ПО точек доступа, когда включен менеджер обновления ПО по расписанию (update-mgr).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
end-time <TIME>  
no end-time
```

## Параметры

<TIME> – окончание интервала времени, в который производится обновление ПО точек доступа, задается в виде HH:MM, где:

- HH – часы, принимает значение [0..23];
- MM – минуты, принимает значение [0..59].

## Значение по умолчанию

04:00

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-UPDATE-MANAGER

```
wlc(config-wlc-update-manager)# end-time 06:00
```

**scheduled**

Данная команда активирует работу менеджера обновления ПО точек доступа по расписанию. Использование отрицательной формы команды (no) отключает менеджер обновления ПО по расписанию.

**Синтаксис**

```
[no] scheduled
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-UPDATE-MANAGER

**Пример**

```
wlc(config-wlc-update-manager)# scheduled
```

**start-time**

Данная команда задает начало временного интервала, в котором производится обновление ПО точек доступа, когда включен менеджер обновления ПО по расписанию (update-mgr).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
start-time <TIME>
no start-time
```

**Параметры**

<TIME> – начало интервала времени, в который производится обновление ПО точек доступа, задается в виде HH:MM, где:

- HH – часы, принимает значение [0..23];
- MM – минуты, принимает значение [0 .. 59].

**Значение по умолчанию**

03:00

**Необходимый уровень привилегий**

10

## Командный режим

CONFIG-WLC-UPDATE-MANAGER

### Пример

```
wlc(config-wlc-update-manager)# start-time 05:00
```

## Настройка резервирования

### failover

Данная команда активирует работу резервирования WLC. Указание адресации и группы описано в разделе [«Резервирование»](#).

Использование отрицательной формы команды (no) выключает работу резервирования WLC.

### Синтаксис

```
[no] failover
```

### Параметры

Отключено.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC

### Пример

```
wlc(config-wlc)# failover
```

## Команды просмотра

### show wlc

Данная команда используется для просмотра общей информации о работе WLC. В выводе содержится общее количество точек доступа; количество точек доступа, находящихся в определенном статусе; общее количество беспроводных клиентов, подключенных ко всем точкам доступа, находящимся под управлением контроллера, и количество клиентов, подключенных в каждом частотном диапазоне; общее количество SSID и количество SSID, работающих в каждом частотном диапазоне.

Описание статусов:

- Active – точка доступа подключена, сконфигурирована и находится в работе;
- Failed – точка доступа отключена или до нее потерян доступ;

- Applying cfg – соединение по протоколу netconf установлено и точка в данный момент применяет конфигурацию, сгенерированную WLC;
- Cfg Failed – конфигурация для точки предоставлена с ошибками. Подробнее можно посмотреть командой `show wlc configuration warnings`;
- Ready – точка доступа содержит актуальную версию программного обеспечения, установила пароль из конфигурации и готова к соединению по протоколу netconf;
- Rebooting – точка доступа перезагружается по запросу администратора;
- Reconnecting – точка доступа прекратила netconf-соединение и пытается снова подключиться;
- Registering – точка доступа прошла регистрацию и получила сертификат;
- Sandboxed – соединение по netconf установлено, но на WLC нет конфигурации для данной точки, поэтому на нее применяется конфигурации по умолчанию;
- Updating creds – в данный момент точка обновляет пароль, netconf-соединение разорвано;
- Upgrading FW – на точке доступа происходит обновление программного обеспечения.

## Синтаксис

```
show wlc
```

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
wlc# sh wlc
  AP Status          Count
  -----
  Active             1
  Failed             0
  Applying cfg       0
  Cfg Failed         0
  Ready              0
  Rebooting          0
  Reconnecting       1
  Registering        0
  Sandboxed          0
  Updating creds     0
  Upgrading FW       0
  -----
  Total              2

  Clients            Count
  -----
  Clients 2g         0
  Clients 5g         0
  -----
  Total              0

  SSIDs              Count
```

-----	-----
SSIDs 2g	1
SSIDs 5g	1
-----	-----
Total	2

## show wlc airtune

Данная команда используется для просмотра информации о работе airtune. В выводе содержится общее количество точек доступа; количество точек доступа, находящихся под управлением airtune, разделенных по ap-location.

Также можно указать MAC-адрес точки доступа для получения детальной информации.

## Синтаксис

```
show wlc airtune [<MAC-AP>]
```

## Параметры

<MAC-AP> – MAC-адрес точки доступа в формате HH:HH:HH:HH:HH:HH

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
wlc# show wlc airtune
AP location                AP airtune  AP count
-----
default-location          2           3
no-airtune                 0           1

wlc# show wlc airtune 68:13:e2:35:d2:20
MAC address: '68:13:e2:35:d2:20':
2g
  MAC radio:                68:13:e2:35:d2:20
  Status:                    Up
  Locked TPC:                 0
  Locked DCA:                 0
  Locked balancer:           1
  Channel:                    1
  Power:                      16
  Max power:                  16
  Min power:                   0
  Bandwidth:                  20
  Available channels:         1,6,11
  SSID:                       TEST-SSID-WLC
  MAC VAP:                    68:13:e2:35:d2:23
  802.11k:                     Enabled
  802.11r:                     Disabled
  802.11v:                     Enabled
```

```

5g
MAC radio:          68:13:e2:35:d2:28
Status:            Up
Locked TPC:        0
Locked DCA:        0
Locked balancer:   1
Channel:           36
Power:             19
Max power:         19
Min power:         0
Bandwidth:         20
Available channels: 36,40,44,48,52,56,60,64
SSID:              TEST-SSID-WLC
  MAC VAP:         68:13:e2:35:d2:2b
  802.11k:         Enabled
  802.11r:         Disabled
  802.11v:         Enabled

```

### show wlc airtune roaming statistics

Данная команда используется для просмотра данных по роумингу в локации. Возможна фильтрация по MAC-адресу точки доступа и/или частотному диапазону.

#### Синтаксис

```
show wlc airtune roaming statistics <AP-LOCATION> [<MAC-AP> [<BAND>] | <BAND>]
```

#### Параметры

<AP-LOCATION> – название существующего профиля локации, задается строкой до 235 символов;

<BAND> – частотный диапазон, допустимы значения: 2g, 5g.

#### Необходимый уровень привилегий

1

#### Командный режим

ROOT

#### Пример

```

wlc# show wlc airtune roaming statistics with-gre

  2g
  ~~
MAC AP          MAC VAP          802.11k  802.11r  802.11v  FBT links  SSID
-----
68:13:e2:35:d2:20 68:13:e2:35:d2:21  Enabled  Disabled  Enabled  0          WLC_OPEN

  5g
  ~~
MAC AP          MAC VAP          802.11k  802.11r  802.11v  FBT links  SSID
-----

```

```
68:13:e2:35:d2:20 68:13:e2:35:d2:29 Enabled Disabled Enabled 0 WLC_OPEN
```

```
wlc# show wlc airtune roaming statistics with-gre 68:13:e2:35:d2:20 2g
```

```
2g
~~
MAC AP          MAC VAP          802.11k  802.11r  802.11v  FBT links  SSID
-----
68:13:e2:35:d2:20 68:13:e2:35:d2:21 Enabled  Disabled  Enabled  0          WLC_OPEN
```

### show wlc airtune rrm optimization report

Данная команда используется для просмотра отчета оптимизации в локации. Можно просмотреть отчет за указанное количество дней до настоящего момента и/или с фильтрацией по частотному диапазону.

#### Синтаксис

```
show wlc airtune rrm optimization report <AP-LOCATION> [<DAYS> [band <BAND>]] band <BAND>]
```

#### Параметры

<AP-LOCATION> – название существующего профиля локации, задается строкой до 235 символов;

<DAYS> – количество дней до настоящего момента, возможны значения от 0;

<BAND> – частотный диапазон, допустимы значения: 2g, 5g.

#### Необходимый уровень привилегий

1

#### Командный режим

ROOT

#### Пример

```
wlc# show wlc airtune rrm optimization report default-location
```

```
Report timezone:      GMT 0
```

```
AP location timezone: GMT +7
```

```
2g
~~
MAC AP          IP address      Board type      AP location      Report time
Power  Power  Channel  Channel
before after  before  after
-----
e8:28:c1:da:f6:40 100.166.1.11  WOP-2L         default-location  22.04.2023 04:22
11      11      1         1
e8:28:c1:fd:c0:a0 100.166.1.13  WEP-3ax        default-location  22.04.2023 04:22
6        6        6         6
```



```

5g
~~
MAC AP          IP address      Board type      AP location      Report time
Power   Power   Channel  Channel
before  after   before   after
-----  -----  -----  -----  -----
e8:28:c1:da:f6:40  100.166.1.11  WOP-2L        default-location  22.04.2023 04:22
19      19      36          36
e8:28:c1:fd:c0:a0  100.166.1.13  WEP-3ax       default-location  22.04.2023 04:22
19      10      44          44

wlc# show wlc airtune rrm optimization report default-location 5 band 5
Report timezone:      GMT 0
AP location timezone: GMT +7

```

```

5g
~~
MAC AP          IP address      Board type      AP location      Report time
Power   Power   Channel  Channel
before  after   before   after
-----  -----  -----  -----  -----
e8:28:c1:da:f6:40  100.166.1.11  WOP-2L        default-location  18.04.2023 07:43
19      19      36          36
e8:28:c1:fd:c0:a0  100.166.1.13  WEP-3ax       default-location  18.04.2023 07:43
11      19      36          36
e8:28:c1:da:f6:40  100.166.1.11  WOP-2L        default-location  19.04.2023 04:22
19      19      36          36
e8:28:c1:fd:c0:a0  100.166.1.13  WEP-3ax       default-location  19.04.2023 04:22
19      19      36          44
e8:28:c1:da:f6:40  100.166.1.11  WOP-2L        default-location  20.04.2023 04:22
19      19      36          36
e8:28:c1:fd:c0:a0  100.166.1.13  WEP-3ax       default-location  20.04.2023 04:22
19      19      44          44
e8:28:c1:da:f6:40  100.166.1.11  WOP-2L        default-location  21.04.2023 04:22
19      19      36          36
e8:28:c1:fd:c0:a0  100.166.1.13  WEP-3ax       default-location  21.04.2023 04:22
19      19      44          44
e8:28:c1:da:f6:40  100.166.1.11  WOP-2L        default-location  22.04.2023 04:22
19      19      36          36
e8:28:c1:fd:c0:a0  100.166.1.13  WEP-3ax       default-location  22.04.2023 04:22
19      10      44          44

```

### show wlc airtune rrm statistics

Данная команда используется для просмотра данных RRM в локации.

### Синтаксис

```
show wlc airtune rrm statistics <AP-LOCATION>
```

### Параметры

<AP-LOCATION> – название существующего профиля локации, задается строкой до 235 символов.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример**

```
wlc# show wlc airtune rrm statistics default-location

  2g
  ~~
MAC AP      MAC radio      Status  Locked  Locked  Channel  Power  Bandwidth
Assoc      Available
clients    channels
-----
e8:28:c1:da:f6:40  e8:28:c1:da:f6:40  Up      0      0      1      11     20
0          1,6,11
e8:28:c1:fd:c0:a0  e8:28:c1:fd:c0:a0  Up      0      0      6      6      20
0          1,6,11

  5g
  ~~
MAC AP      MAC radio      Status  Locked  Locked  Channel  Power  Bandwidth
Assoc      Available
clients    channels
-----
e8:28:c1:da:f6:40  e8:28:c1:da:f6:48  Up      0      0      36     19     20
0          36,40,44,
48,52,56,
60,64,132
e8:28:c1:fd:c0:a0  e8:28:c1:fd:c0:b0  Up      0      0      44     10     20
0          36,40,44,
48
```

**show wlc airtune sessions**

Данная команда используется для просмотра активных сессий в airtune в указанной локации.

**Синтаксис**

```
show wlc airtune sessions <AP-LOCATION>
```

**Параметры**

<AP-LOCATION> – название существующего профиля локации, задается строкой до 235 символов.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример**

```
wlc# show wlc airtune sessions default-location
MAC AP                IP address           Board type           Session ID           AP location
-----
e8:28:c1:da:f6:40    100.166.1.11        WOP-2L              1                    default-location
e8:28:c1:fd:c0:a0    100.166.1.13        WEP-3ax             2                    default-location
```

**show wlc configuration warnings**

Данная команда используется для просмотра предупреждений о проблемах, связанных с применением заданной конфигурации на точке доступа.

**Синтаксис**

```
show wlc configuration warnings
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример**

```
wlc# show wlc configuration warnings
No      Message
-----
0       Cannot apply ssid 'test' to radio 'wlan0' on
        'cc:9d:a2:c7:c6:50'
1       Cannot apply ssid 'test' to radio 'wlan1' on
        'cc:9d:a2:c7:c6:50'
```

**show wlc ap**

Данная команда используется для просмотра информации о точках доступа, обслуживаемых контроллером. Вывод содержит MAC-адрес, IP-адрес точки доступа, модель точки доступа, версию программного обеспечения, статус, количество клиентов и другую информацию.

Описание выводимых параметров:

- MAC address – MAC-адрес – MAC-адрес точки доступа;
- Status – Статус – состояние точки доступа;
- Status description – Описание статуса – дополнительная информация по статусу;
- IP address – IP-адрес – IP-адрес точки доступа;
- Board type – Модель – модель точки доступа;
- SW version – Версия ПО – версия программного обеспечения точки доступа;
- Serial number – Серийный номер – серийный номер устройства, установленный заводом-изготовителем;
- HW version – Аппаратная версия – HW-версия аппаратного обеспечения устройства;
- First activity at – Первая активность – время первой регистрации точки доступа на контроллере;
- Connected at – Подключен в – время последнего подключения точки доступа к контроллеру;
- Last activity at – Последняя активность – время, в которое контроллер последний раз настраивал точку доступа;
- Clients 2g – Количество клиентов в 2.4 ГГц – число клиентов в 2.4 ГГц, подключенных к точке доступа;
- Clients 5g – Количество клиентов в 5 ГГц – число клиентов в 5 ГГц, подключенных к точке доступа;
- Clients all – Количество клиентов – общее число клиентов, подключенных к точке доступа;
- Hostname – Имя хоста – сетевое имя точки доступа;
- Ap-location – Локация – локация, в которой находится точка доступа;
- Configured – Подключена через – профиль, с помощью которого точка доступа была настроена – ip pool или mac;
- Netconf connection state – Состояние Netconf – статус соединения точки доступа и контроллера по протоколу Netconf;
- Description – Описание – текстовое описание точки доступа, которое ей было назначено при формировании профиля;
- Uptime (d,h:m:s) – Время работы – время работы с момента последнего включения или перезагрузки устройства;
- Band – Диапазон, ГГц – частотный диапазон радиointерфейса;
- MAC address – MAC-адрес радиointерфейса – MAC-адрес радиointерфейса точки доступа;
- Status – Статус радиointерфейса – состояние радиointерфейса;
- Channel – Номер канала – номер беспроводного канала, на котором работает радиointерфейс;
- Frequency – Частота, МГц – частота, на которой работает радиointерфейс;
- Bandwidth – Ширина канала, МГц – ширина полосы частот канала, на которой работает радиointерфейс;
- TX power – Мощность, дБм – мощность сигнала радиointерфейса;
- Utilization – Утилизация канала – процент времени, в течение которого канал используется для передачи трафика;
- Average utilization – Средняя утилизация канала – средний показатель утилизации канала за 1 час;
- Max utilization – Максимальная утилизация канала – максимальный показатель утилизации канала за 1 час;

Описание статусов:

- Active – точка доступа подключена, сконфигурирована и находится в работе;
- Failed – точка доступа отключена или до нее потерян доступ;
- Applying cfg – соединение по протоколу netconf установлено и точка в данный момент применяет конфигурацию, сгенерированную WLC;
- Cfg Failed – конфигурация для точки предоставлена с ошибками. Подробнее можно посмотреть командой [show wlc configuration warnings](#);

- Ready – точка доступа содержит актуальную версию программного обеспечения, установила пароль из конфигурации и готова к соединению по протоколу netconf;
- Rebooting – точка доступа перезагружается по запросу администратора;
- Reconnecting – точка доступа прекратила netconf-соединение и пытается снова подключиться;
- Registering – точка доступа прошла регистрацию и получила сертификат;
- Sandboxed – соединение по netconf установлено, но на WLC нет конфигурации для данной точки, поэтому на нее применяется конфигурации по умолчанию;
- Updating creds – в данный момент точка обновляет пароль, netconf-соединение разорвано;
- Upgrading FW – на точке доступа происходит обновление программного обеспечения.

## Синтаксис

```
show wlc ap [<AP-MAC>] [detailed] [detailed <AP-MAC>]
```

## Параметры

<AP-MAC> – MAC-адрес точки доступа. При указании данного параметра будет отображена информация только по заданной точке доступа.

detailed – просмотр подробной информации по одной или всем точкам доступа.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
wlc# show wlc ap
MAC address      Status      IP address      SW version      Hostname
Ap-location      Uptime      Clients(2g/5g/all)
-----
cc:9d:a2:c7:c6:50 Failed      192.168.1.2     1.14.1 build 3  WEP-3ax
default-location --          0/0/0
e0:d9:e3:48:aa:80 Sandboxed   192.168.1.5     1.25.2.25       WEP-2ac_Smart
00,00:01:00      0/0/0
e8:28:c1:e1:cf:e0 Active      192.168.1.4     2.6.0 build 592 WOP-2L
location2        01,20:17:04    0/0/0

wlc# show wlc ap detailed e8:28:c1:e1:cf:e0
AP e8:28:c1:e1:cf:e0:
  MAC address:      e8:28:c1:e1:cf:e0
  Status:           Active
  Status description: --
  IP address:       192.168.1.4
  Board type:       WOP-2L
  SW version:       2.6.0 build 592
  Serial number:    WP39000149
  HW version:       1v2
  First activity at: 2024.11.29 03:24
  Connected at:     2024.12.03 06:30
  Last activity at: 2024.12.03 07:40
  Clients 2g:       0
```

```

Clients 5g:           0
Clients all:         0
Hostname:            WOP-2L
Ap-location:         location2
Configured as:       ip-pool pool2
Netconf connection state: Alive
Description:
Uptime (d,h:m:s):    01,20:19:37
Radio wlan0:
Band:                2.4
MAC address:         e8:28:c1:e1:cf:e0
Status:              Enabled
Channel:             11
Frequency:           2462
Bandwidth:           20
TX power:            16
Utilization:         68
Average utilization: 71
Max utilization:     87
Radio wlan1:
Band:                5
MAC address:         e8:28:c1:e1:cf:e8
Status:              Enabled
Channel:             48
Frequency:           5240
Bandwidth:           20
TX power:            19
Utilization:         18
Average utilization: 16
Max utilization:     56

```

## show wlc ap interfaces

Данная команда используется для просмотра информации и счетчиков по интерфейсам точки доступа.

### Синтаксис

```
show wlc ap interfaces <AP-MAC> [<IF-NAME>]
```

### Параметры

<AP-MAC> – MAC-адрес точки доступа в формате НН:НН:НН:НН:НН:НН. Будет выведена информация и статистика по интерфейсам указанной точки доступа;

<IF-NAME> – название интерфейса точки доступа, задается строкой до 15 символов. При указании данного параметра будет отображена статистика только по указанному интерфейсу.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

**Пример**

```
wlc# show wlc ap interfaces e8:28:c1:da:c9:b0
```

Interface	MAC address	State	RX bytes	RX packets	RX errors	RX drops
br0	e8:28:c1:da:c9:b0	Up	0	0	0	0
eth0	e8:28:c1:da:c9:b0	Up	33691	299	0	0
lsw	e8:28:c1:da:c9:b1	Up	0	0	0	0
user-gre	e8:28:c1:da:c9:b2	Up	0	0	0	0
wlan0	e8:28:c1:da:c9:b0	Up	0	0	0	0
wlan0-va0	e8:28:c1:da:c9:b1	Up	0	0	0	0
wlan0-va1	00:e0:4c:81:86:86	Down	0	0	0	0
wlan0-va2	00:e0:4c:81:86:86	Down	0	0	0	0
wlan0-va3	00:e0:4c:81:86:86	Down	0	0	0	0
wlan1	e8:28:c1:da:c9:b5	Up	0	0	0	0
wlan1-va0	e8:28:c1:da:c9:b6	Up	0	0	0	0
wlan1-va1	00:e0:4c:81:86:86	Down	0	0	0	0
wlan1-va2	00:e0:4c:81:86:86	Down	0	0	0	0
wlan1-va3	00:e0:4c:81:86:86	Down	0	0	0	0

Interface	MAC address	State	TX bytes	TX packets	TX errors	TX drops
br0	e8:28:c1:da:c9:b0	Up	0	0	0	0
eth0	e8:28:c1:da:c9:b0	Up	88245	273	0	0
lsw	e8:28:c1:da:c9:b1	Up	0	0	0	0
user-gre	e8:28:c1:da:c9:b2	Up	0	0	0	0
wlan0	e8:28:c1:da:c9:b0	Up	0	0	0	0
wlan0-va0	e8:28:c1:da:c9:b1	Up	0	0	0	0
wlan0-va1	00:e0:4c:81:86:86	Down	0	0	0	0
wlan0-va2	00:e0:4c:81:86:86	Down	0	0	0	0
wlan0-va3	00:e0:4c:81:86:86	Down	0	0	0	0
wlan1	e8:28:c1:da:c9:b5	Up	0	0	0	0
wlan1-va0	e8:28:c1:da:c9:b6	Up	0	0	0	0
wlan1-va1	00:e0:4c:81:86:86	Down	0	0	0	0
wlan1-va2	00:e0:4c:81:86:86	Down	0	0	0	0
wlan1-va3	00:e0:4c:81:86:86	Down	0	0	0	0

```
wlc# show wlc ap interfaces e8:28:c1:da:c9:b0 wlan0-va0
```

```
Interface wlan0-va0:
```

```
  MAC address:          e8:28:c1:da:c9:b1
  Operational state:    Up
  Link speed:           144444444
  Duplex:                unavailable
  Actual transmit rate: 0
  Actual receive rate:  0
  Transmit bytes:       0
  Receive bytes:        0
  Transmit packets:    0
  Receive packets:     0
  Transmit errors:     0
  Receive errors:      0
  Transmit drops:      0
  Receive drops:       0
```

**show wlc ap radios**

Данная команда используется для просмотра основных параметров радиоинтерфейсов точки доступа.

## Синтаксис

```
show wlc ap radios <AP-MAC> [<IF-NAME>]
```

## Параметры

<AP-MAC> – MAC-адрес точки доступа в формате HH:HH:HH:HH:HH:HH. Будет выведена информация о радиointерфейсах указанной точки доступа;

<IF-NAME> – название радиointерфейса точки доступа, задается строкой до 15 символов. При указании данного параметра будет отображена информация только по указанному радиointерфейсу.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
wlc# show wlc ap radios e8:28:c1:da:c9:b0
Radio wlan0:
  Band:                2.4
  MAC address:         e8:28:c1:da:c9:b0
  Status:              Enabled
  Channel:             11
  Frequency:           2462
  Bandwidth:           20
  TX power:            16
-----
Radio wlan1:
  Band:                5
  MAC address:         e8:28:c1:da:c9:b5
  Status:              Enabled
  Channel:             48
  Frequency:           5240
  Bandwidth:           20
  TX power:            19

wlc# show wlc ap radios e8:28:c1:da:c9:b0 wlan1
Radio wlan1:
  Band:                5
  MAC address:         e8:28:c1:da:c9:b5
  Status:              Enabled
  Channel:             48
  Frequency:           5240
  Bandwidth:           20
  TX power:            19
```

## show wlc ap vap

Данная команда используется для просмотра основных параметров виртуальных точек доступа (VAP) точки доступа.



## Синтаксис

```
show wlc ap vap [<AP-MAC> | ap-location <AP-LOCATION>]
```

## Параметры

<AP-MAC> – MAC-адрес точки доступа в формате HH:HH:HH:HH:HH:HH. Будет выведена информация о VAP указанной точки доступа;

<AP-LOCATION> – название профиля локации, задается строкой до 235 символов. При указании данного параметра будет отображена информация только по указанной локации.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
wlc# sh wlc ap vap
```

MAC AP	Hostname	RadioID	VapID	SSID	BSSID
Auth type	Clients				
e8:28:c1:00:27:11	ap_414_wep	0	0	default-ssid30	
e8:28:c1:00:27:12	WPA2 Enterprise	0			
e8:28:c1:00:27:11	ap_414_wep	0	1	default-ssid_off	
e8:28:c1:00:27:13	Open	5			
e8:28:c1:00:27:11	ap_414_wep	1	0	default-ssid30	
e8:28:c1:00:27:19	WPA2 Enterprise	0			
e8:28:c1:00:27:11	ap_414_wep	1	1	ssid_WPA	
e8:28:c1:00:27:1a	WPA PSK	0			
e8:28:c1:00:27:11	ap_414_wep	1	2	ssid_WPA2_1X	
e8:28:c1:00:27:1b	WPA2 Enterprise	0			
e8:28:c1:00:27:11	ap_414_wep	1	3	ssid_WPA2_WPA3	
e8:28:c1:00:27:1c	WPA2 PSK	2			
e8:28:c1:00:27:11	ap_414_wep	1	4	ssid_WPA3	
e8:28:c1:00:27:1d	WPA2 PSK	0			
e8:28:c1:00:27:11	ap_414_wep	1	5	WPA2_WPA3_1X	
e8:28:c1:00:27:1e	WPA2 Enterprise	0			
e8:28:c1:00:27:13	--	0	0	default-ssid30	
e8:28:c1:00:27:14	WPA2 Enterprise	0			
e8:28:c1:00:27:13	--	0	1	default-ssid_off	
e8:28:c1:00:27:15	Open	0			
e8:28:c1:00:27:13	--	1	0	default-ssid30	
e8:28:c1:00:27:1b	WPA2 Enterprise	0			
e8:28:c1:00:27:13	--	1	1	ssid_WPA	
e8:28:c1:00:27:1c	WPA PSK	0			
e8:28:c1:00:27:13	--	1	2	ssid_WPA2_1X	
e8:28:c1:00:27:1d	WPA2 Enterprise	0			
e8:28:c1:00:27:13	--	1	3	ssid_WPA2_WPA3	
e8:28:c1:00:27:1e	WPA2 PSK	0			
e8:28:c1:00:27:13	--	1	4	ssid_WPA2_WPA3	
e8:28:c1:00:27:1f	WPA2 PSK	0			

```

e8:28:c1:00:27:13  --          1          5          WPA2_WPA3_1X
e8:28:c1:00:27:20  WPA2 Enterprise    0

wlc# show wlc ap vap e8:28:c1:00:27:11
Hostname          RadioID  VapID  SSID          BSSID          Auth
type             Clients
-----
ap_414_wep        0        0      default-ssid30  e8:28:c1:00:27:12  WPA2
Enterprise        0
ap_414_wep        0        1      default-ssid_off  e8:28:c1:00:27:13  Open
5
ap_414_wep        1        0      default-ssid30  e8:28:c1:00:27:19  WPA2
Enterprise        0
ap_414_wep        1        1      ssid_WPA        e8:28:c1:00:27:1a  WPA
PSK                0
ap_414_wep        1        2      ssid_WPA2_1X    e8:28:c1:00:27:1b  WPA2
Enterprise        0
ap_414_wep        1        3      ssid_WPA2_WPA3  e8:28:c1:00:27:1c  WPA2
PSK                0
ap_414_wep        1        4      ssid_WPA3       e8:28:c1:00:27:1d  WPA2
PSK                2
ap_414_wep        1        5      WPA2_WPA3_1X    e8:28:c1:00:27:1e  WPA2
Enterprise        0

wlc# sh wlc ap vap ap-location default-location
MAC AP           Hostname          RadioID  VapID  SSID          BSSID
Auth type       Clients
-----
e8:28:c1:00:28:18  ap_515_wep        0        0      default-ssid30
e8:28:c1:00:27:12  WPA2 Enterprise    0
e8:28:c1:00:28:18  ap_515_wep        0        1      default-ssid_off
e8:28:c1:00:27:13  Open              10
e8:28:c1:00:28:18  ap_515_wep        1        0      default-ssid30
e8:28:c1:00:27:19  WPA2 Enterprise    0
e8:28:c1:00:28:18  ap_515_wep        1        1      ssid_WPA
e8:28:c1:00:27:1a  WPA PSK           0
e8:28:c1:00:28:18  ap_515_wep        1        2      ssid_WPA2_1X
e8:28:c1:00:27:1b  WPA2 Enterprise    0
e8:28:c1:00:28:18  ap_515_wep        1        3      ssid_WPA2_WPA3
e8:28:c1:00:27:1c  WPA2 PSK          0
e8:28:c1:00:28:18  ap_515_wep        1        4      ssid_WPA3
e8:28:c1:00:27:1d  WPA2 PSK          0
e8:28:c1:00:28:18  ap_515_wep        1        5      WPA2_WPA3_1X
e8:28:c1:00:27:1e  WPA2 Enterprise    1

```

### show wlc ap firmware

Данная команда используется для просмотра информации о минимальной поддерживаемой версии ПО для каждой модели точки доступа, а также о загруженных на WLC актуальных файлах ПО ТД с указанием названия файла и версии ПО. Если для одной модели ТД загружено более одного файла, то в графе "Current SW version" отображается самая актуальная версия ПО (старшая по номеру). Все загруженные файлы ПО ТД на контроллер можно увидеть с помощью команды: `dir system:access-points-firmwares`.

### Синтаксис

```
show wlc ap firmware
```

## Параметры

Отсутствуют.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

Команда, показывает, что на контроллер загружены файлы ПО точек доступа: WEP-200L, WEP-30L, WEP-30L-Z. Версии ПО ТД и названия файлов ПО ТД отображаются в столбцах Current SW version и Filename соответственно.

```
wlc# show wlc ap firmware
```

Board type Filename	Min SW version	Current SW version
-----	-----	
-----		-----
WEP-1L	2.5.6 build 0	--
--		
WEP-200L	2.6.0 build 0	2.6.0 build 762
WEP-200L-2.6.0_build_762.tar.gz		
WEP-2L	2.5.6 build 0	--
--		
WEP-2ac	1.25.2 build 0	--
--		
WEP-2ac Smart	1.25.2 build 0	--
--		
WEP-30L	2.6.0 build 0	2.6.2 build 50
WEP-30L-2.6.2_build_50.tar.gz		
WEP-30L-Z	2.6.0 build 0	2.6.2 build 50
WEP-30L-2.6.2_build_50.tar.gz		
WEP-3L	2.5.3 build 0	--
--		
WEP-3ax	1.14.0 build 0	--
--		
WOP-20L	2.6.0 build 0	--
--		
WOP-2L	2.5.6 build 0	--
--		
WOP-2ac	1.25.2 build 0	--
--		
WOP-2ac:rev.B	1.25.2 build 0	--
--		
WOP-2ac:rev.C	1.25.2 build 0	--
--		
WOP-30L	2.6.0 build 0	--
--		
WOP-30LI	2.6.0 build 0	--
--		
WOP-30LS	2.6.0 build 0	--
--		

## show wlc clients

Данная команда используется для просмотра информации о подключенных к контроллеру беспроводных клиентах. Возможна фильтрация по локации, MAC-адресу клиента или точки доступа.

### Синтаксис

```
show wlc clients [ <CLIENT-MAC> [detailed] | ap <AP-MAC> [detailed] | ap-location <AP-LOCATION> [detailed]]
```

### Параметры

<CLIENT-MAC> – MAC-адрес беспроводного клиента в формате HH:HH:HH:HH:HH:HH. При указании данного параметра будет отображена информация только об указанном клиенте.

<AP-MAC> – MAC-адрес точки доступа в формате HH:HH:HH:HH:HH:HH. Будет выведен список клиентов указанной точки доступа;

detailed – просмотр подробной информации о всех беспроводных клиентах указанной точки доступа или указанного клиента;

<AP-LOCATION> – имя AP-Location. Будет выведен список клиентов для точек доступа данной локации.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
wlc# show wlc clients
MAC User          IP User          MAC AP          Hostname AP          SSID
RSSI      AP-Location      Username
-----
6a:83:2a:48:0d:72  100.139.52.4    68:13:e2:35:d2:20  WEP-30L-TEST
TEST-SSID-WLC-15  -38            with-gre          tester

wlc# show wlc clients ap 68:13:e2:35:d2:20
MAC address      IP address      SSID          Uptime      RSSI      SNR      Mode
LQ      Username
-----
6a:83:2a:48:0d:72  100.139.52.4    TEST-SSID-WLC-15  00:01:48    -42 -35    35 35    ac
100    tester

wlc# show wlc clients ap e4:5a:d4:f0:6e:b0 detailed
Client client-1:
  MAC address:      d8:c7:71:c9:d8:19
  IP address:       100.166.2.2
  Hostname:         DLI-TL20
  SSID:             MTV_ENTER
  Interface:        wlan0-va0
  RSSI:             -39 -32
```

```

SNR:                13 14
Transmit rate:      MCS4 NO SGI 39
Receive rate:       MCS6 NO SGI 58.5
Wireless mode:      n
Authorized:         true
Username:           tester123
Domain:             wlc.root
Link Quality:       90
Link Quality Common: 85
Actual transmit rate: 476
Actual receive rate: 86
Transmit bytes:     27739443
Receive bytes:      1934683
Transmit packets:   21663
Receive packets:    11402
Uptime:             00:02:02
Transmit bandwidth: 20M
Receive bandwidth:  20M

```

## show wlc journal ap

Данная команда используется для просмотра информации о процессе подключения к контроллеру, обновлении и конфигурировании точек доступа.

### Синтаксис

```

show wlc journal ap [<AP-MAC>] [ from-date <YEAR> <MONTH> <DAY> ] [ from-time <TIME> ]
[ to-date <YEAR> <MONTH> <DAY> ] [ to-time <TIME> ] [ from-end ]

```

### Параметры

<AP-MAC> – MAC-адрес точки доступа в формате HH:HH:HH:HH:HH:HH. Будет выведен журнал с информацией об указанной точке доступа;

from-date – вывод информации будет осуществляться начиная с указываемой даты;

from-time – вывод информации будет осуществляться начиная с указываемого времени;

to-date – вывод информации будет осуществляться по указываемую дату (включительно);

to-time – вывод информации будет осуществляться по указываемое время (включительно);

<YEAR> – год, принимает значения [2001..2037];

<MONTH> – месяц, принимает значения [January/February/March/April/May/June/July/August/September/October/November/December];

<DAY> – день месяца, принимает значения [1..31];

<TIME> – время, задается в виде HH MM SS, где:

- HH – часы, принимает значение [0..23];
- MM – минуты, принимает значение [0 .. 59];
- SS – секунды, принимает значение [0 .. 59].

from-end – просмотр содержимого файла будет осуществляться с конца, так как последние записи помещаются в конец файла.

### Необходимый уровень привилегий

1

## Командный режим

ROOT

### Пример

```
wlc# show wlc journal ap
2022-08-11T15:28:10+00:00 AP e8:28:c1:da:c9:b0 changed state to 'Registering'
2022-08-11T15:28:11+00:00 AP e8:28:c1:da:c9:b0 changed state to 'Registering'
2022-08-11T15:28:12+00:00 AP e8:28:c1:da:c9:b0 changed state to 'Ready'
2022-08-11T15:28:13+00:00 AP e8:28:c1:da:c9:b0 connected, board 'WEP-1L' sw version '1.2.6
build 5' ipaddr '192.168.1.2'
2022-08-11T15:28:13+00:00 AP e8:28:c1:da:c9:b0 changed state to 'Applying cfg'
2022-08-11T15:28:13+00:00 AP e8:28:c1:da:c9:b0 changed state to 'Active'
```

### show wlc journal clients

Данная команда используется для просмотра журнала событий клиентов: подключения, отключения, ошибки авторизации, роуминг.

### Синтаксис

```
show wlc journal clients [<CLIENT-MAC>] [ from-date <YEAR> <MONTH> <DAY> ] [ from-time
<TIME> ] [ to-date <YEAR> <MONTH> <DAY> ] [ to-time <TIME> ] [ from-end ]
```

### Параметры

<CLIENT-MAC> – MAC-адрес беспроводного клиента в формате HH:HH:HH:HH:HH:HH. При указании данного параметра будет отображена информация в журнале только об указанном клиенте;

from-date – вывод информации будет осуществляться начиная с указываемой даты;

from-time – вывод информации будет осуществляться начиная с указываемого времени;

to-date – вывод информации будет осуществляться по указываемую дату (включительно);

to-time – вывод информации будет осуществляться по указываемое время (включительно);

<YEAR> – год, принимает значения [2001..2037];

<MONTH> – месяц, принимает значения [January/February/March/April/May/June/July/August/September/October/November/December];

<DAY> – день месяца, принимает значения [1..31];

<TIME> – время, задается в виде HH MM SS, где:

- HH – часы, принимает значение [0..23];
- MM – минуты, принимает значение [0 .. 59];
- SS – секунды, принимает значение [0 .. 59].

from-end – просмотр содержимого файла будет осуществляться с конца, так как последние записи помещаются в конец файла.

### Необходимый уровень привилегий

1

## Командный режим

ROOT

### Пример

```
wlc# show wlc journal clients
2022-09-14T13:30:17+00:00 client 60:ab:67:ba:89:24 Failed on AP cc:9d:a2:c7:c6:50 SSID test
reason 69, 'Incorrect password'
2022-09-14T13:30:34+00:00 client 60:ab:67:ba:89:24 Connected on AP cc:9d:a2:c7:c6:50 SSID test
RSSI: -46,-48
2022-09-14T13:30:52+00:00 client 60:ab:67:ba:89:24 Disconnected on AP cc:9d:a2:c7:c6:50 SSID
test reason 3, 'Sending station is leaving (or has left) IBSS or ESS'
```

### show wlc journal wids

Данная команда используется для просмотра журнала событий WIDS.

### Синтаксис

```
show wlc journal wids [ <SOURCE-AP-MAC> | ap-location <AP-LOCATION> | attack <ATTACK-
NAME> [<ATTACK-EVENT-TYPE>] ] [ from-date <YEAR> <MONTH> <DAY> ] [ from-time <TIME> ]
[ to-date <YEAR> <MONTH> <DAY> ] [ to-time <TIME> ] [ from-end ]
```

### Параметры

<SOURCE-AP-MAC> – MAC-адрес точки доступа в формате HH:HH:HH:HH:HH:HH. Будет выведен журнал с информацией об атаках, обнаруженных указанной точкой доступа;

<AP-LOCATION> – название локации. Будет выведен журнал для точек доступа данной локации;

<ATTACK-NAME> – название атаки WIDS. Будет выведен журнал с записями только по данному виду атаки. Возможные значения:

- bruteforce – тип атаки Bruteforce (перебор паролей);
- dos – тип атаки DoS (отказ в обслуживании);
- mitm – тип атаки MITM (имитация SSID или MAC-адреса).

<ATTACK-EVENT-TYPE> – тип сообщения об атаке. Будет выведен журнал с фильтром по указанному типу сообщений для определенного вида атаки. Возможные значения в зависимости от вида атаки:

- для атаки типа Bruteforce:
  - too-much-authentication-fail-detected – сообщения, информирующие о слишком большом количестве неудачных попыток авторизации на ТД. Является сигналом о возможной атаке по перебору паролей;
  - client-was-banned – сообщения о том, что клиент был забанен по причине слишком большого количества неудачных попыток авторизации;
  - client-was-unbanned – сообщения о том, что клиент был разбанен по истечению времени блокировки.
- для атаки типа MITM:
  - detected-rogue – сообщения, информирующие об обнаружении "вредоносной" ТД;
  - lost-rogue – сообщения, информирующие о потере из поля видимости "вредоносной" ТД;
  - detected-unknown – сообщения, информирующие об обнаружении неизвестной ТД. Данная ТД не производит каких-либо подозрительных действий. Сообщение генерируется только при включении prevention-mode all;

- `our-attacks-information-update` – сообщения со статистикой об атаках, проводимых нашей ТД относительно стоящих рядом ТД (в зависимости от выбранного `prevention-mode`). Сообщение отсылается только если в данный момент есть ТД, которые нужно атаковать.

`from-date` – вывод информации будет осуществляться начиная с указываемой даты;

`from-time` – вывод информации будет осуществляться начиная с указываемого времени;

`to-date` – вывод информации будет осуществляться по указываемую дату (включительно);

`to-time` – вывод информации будет осуществляться по указываемое время (включительно);

`<YEAR>` – год, принимает значения [2001..2037];

`<MONTH>` – месяц, принимает значения [January/February/March/April/May/June/July/August/September/October/November/December];

`<DAY>` – день месяца, принимает значения [1..31];

`<TIME>` – время, задается в виде HH MM SS, где:

- HH – часы, принимает значение [0..23];
- MM – минуты, принимает значение [0 .. 59];
- SS – секунды, принимает значение [0 .. 59].

`from-end` – просмотр содержимого журнала будет осуществляться с конца, то есть с более новых записей.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
wlc# sh wlc journal wids
2024-11-14T02:24:40-10:00 AP 68:13:e2:20:a2:70 detected DoS attack on wlan0: a sharp increase
in the number of packets with type 'Beacon' (from 39342 to 46336), AP location: test_wids.
Found 1 attacks for last detection period, attack time from AP: 2024-11-14T15:24:40+03:00
2024-11-15T09:16:32-10:00 AP 68:13:e2:20:a2:70 detected DoS attack on wlan0: too many packets
with type 'Beacon' (count 11484 when constraint 11114), AP location: test_wids. Found 87
attacks for last detection period, attack time from AP: 2024-11-15T22:07:42+03:00
```

## `show wlc service-activator aps`

Данная команда используется для просмотра точек доступа в очереди на регистрацию.

Описание статусов:

- `registering` – точка доступа подключилась к сервис-активатору (проверка версии программного обеспечения);
- `awaited` – точка доступа ожидает регистрацию;
- `join-by-admin` – регистрация по команде администратора;
- `join-auto` – регистрация в автоматическом режиме;
- `joined` – точка доступа зарегистрирована;
- `upgrading fw` – точка доступа находится в процессе обновления программного обеспечения.



**Синтаксис**

```
show wlc service-activator aps
```

**Параметры**

Отсутствуют.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример**

```
wlc# show wlc service-activator aps
```

MAC address	Status	IP address	Board type	SW version
HW version	Serial number			
16:79:b2:61:fa:70	Registring	100.124.49.93	WEP-200L	1.6.2 build 11
1v1	WP4B3850143			
72:59:d0:ec:56:a9	Awaited	100.124.49.96	WEP-200L	1.6.2 build 11
1v1	WP4B4367838			
a2:2f:d8:63:23:35	Join-by-admin	100.124.49.97	WEP-200L	1.6.2 build 11
1v1	WP4B7731186			
a6:d6:6f:48:07:78	Join-auto	100.124.49.95	WEP-200L	1.6.2 build 11
1v1	WP4B1435354			
a6:d6:6f:52:57:24	Joined	100.124.49.94	WEP-200L	1.6.2 build 11
1v1	WP4B3465466			
a6:d6:6f:46:07:74	Upgrading_FW	100.124.49.92	WEP-200L	1.5.0 build 15
1v1	WP4B4577656			

**Команды управления****clear wlc airtune session**

Данная команда используется для очистки сессий airtune в локации.

**Синтаксис**

```
clear wlc airtune session [<MAC-AP>]
```

**Параметры**

<MAC-AP> – MAC-адрес точки доступа в формате НН:НН:НН:НН:НН:НН.

**Необходимый уровень привилегий**

10

## Командный режим

ROOT

### Пример

```
wlc# clear wlc airtune session cc:9d:a2:dd:87:50
```

### clear wlc ap

Данная команда используется для удаления точек доступа.

### Синтаксис

```
clear wlc ap [failed | joined [<MAC-AP>] | <MAC-AP>]
```

### Параметры

<MAC-AP> – MAC-адрес точки доступа в формате HH:HH:HH:HH:HH:HH;

failed – удаление всех точек доступа, находящихся в статусе "Failed";

joined – отзыв сертификатов для всех точек или одной по указанному MAC-адресу.

### Необходимый уровень привилегий

10

## Командный режим

ROOT

### Пример

```
wlc# clear wlc ap
```

### clear wlc client

Данная команда используется для отключения клиентских устройств от точек доступа.

### Синтаксис

```
clear wlc client <MAC-CLIENT>
```

### Параметры

<MAC-CLIENT> – MAC-адрес подключенного клиентского устройства в формате HH:HH:HH:HH:HH:HH.

### Необходимый уровень привилегий

10

## Командный режим

ROOT

### Пример

```
wlc# clear wlc client 00:00:00:00:00:00
```

### clear wlc journal

Данная команда используется для удаления журналов событий точек доступа и клиентов.

### Синтаксис

```
clear wlc journal {ap | clients | wids}
```

### Параметры

ap – удаление журнала событий точек доступа;

clients – удаление журнала событий клиентов;

wids – удаление журнала событий WIDS.

### Необходимый уровень привилегий

10

## Командный режим

ROOT

### Пример

```
wlc# clear wlc journal ap
```

### join wlc ap

Данная команда генерирует клиентские сертификаты и регистрирует одну или все точки доступа на WLC.

### Синтаксис

```
join wlc ap [<MAC-AP>]
```

### Параметры

<MAC-AP> – MAC-адрес точки доступа в формате HH:HH:HH:HH:HH:HH.

### Необходимый уровень привилегий

10

**Командный режим**

ROOT

**Пример**

```
wlc# join wlc ap
```

**reload wlc airtune rrm optimization**

Данная команда используется для перезагрузки модуля оптимизации airtune в локации. В результате перезагрузки модуля произойдет оптимизация.

**Синтаксис**

```
reload wlc airtune rrm optimization <AP-LOCATION>
```

**Параметры**

<AP-LOCATION> – название существующего профиля локации, задается строкой до 235 символов.

**Необходимый уровень привилегий**

10

**Командный режим**

ROOT

**Пример**

```
wlc# reload wlc airtune rrm optimization default-location
```

**reload wlc ap**

Данная команда используется для перезагрузки точки доступа, подключенной к контроллеру.

**Синтаксис**

```
reload wlc ap <MAC-AP>
```

**Параметры**

<MAC-AP> – MAC-адрес точки доступа в формате НН:НН:НН:НН:НН:НН.

**Необходимый уровень привилегий**

10

## Командный режим

ROOT

### Пример

```
wlc# reload wlc ap cc:9d:a2:c7:c6:50
```

## set wlc indication enable ap

Данная команда используется для активации индикации светодиодом на ТД, возможно временное включение.

### Синтаксис

```
set wlc indication enable ap <MAC-AP> [timeout <TIMEOUT>]
```

### Параметры

<MAC-AP> – MAC-адрес точки доступа в формате НН:НН:НН:НН:НН:НН.

<TIMEOUT> – временной интервал в секундах, в течение которого будет активна индикация на точке доступа. Принимает значения от 10 до 86400.

### Необходимый уровень привилегий

10

## Командный режим

ROOT

### Пример

```
wlc# set wlc indication enable ap e4:5a:d4:f0:6e:b0 timeout 30
```

## set wlc indication disable ap

Данная команда используется для деактивации индикации светодиодом на ТД.

### Синтаксис

```
set wlc indication disable ap <MAC-AP>
```

### Параметры

<MAC-AP> – MAC-адрес точки доступа в формате НН:НН:НН:НН:НН:НН.

### Необходимый уровень привилегий

10

## Командный режим

ROOT

### Пример

```
wlc# set wlc indication disable ap e4:5a:d4:f0:6e:b0
```

## Настройка логирования на точках доступа

### autolog address

Данной командой задается IP-адрес TFTP-сервера для автоматической выгрузки логов с точки доступа. Использование отрицательной формы команды (no) удаляет IP-адрес TFTP-сервера для автовыгрузки логов.

### Синтаксис

```
address <ADDR>  
no address
```

### Параметры

<ADDR> – IP-адрес TFTP-сервера задается в виде: AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

## Командный режим

config-wlc-ap-profile

### Пример

```
wlc(config-wlc-ap-profile)# address 192.168.1.1
```

### autolog enable

Данная команда включает автовыгрузку логов с точки доступа. Использование отрицательной формы команды (no) отключает автовыгрузку.

### Синтаксис

```
[no] enable
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE

## Пример

```
wlc(config-wlc-ap-profile)# enable
```

## autolog logfile-limit

Данная команда устанавливает пороговое значение размера папки с логами на точке доступа (/var/log/), при превышении которого логи будут автоматически выгружены на заданный TFTP-сервер при включенной опции autolog.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
logfile-limit <LIMIT>  
no logfile-limit
```

## Параметры

<LIMIT> – пороговое значение размера папки с логами на точке доступа, принимает значение в диапазоне [0..20000] КБ.

## Значение по умолчанию

10000

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE

## Пример

```
wlc(config-wlc-ap-profile)# logfile-limit 5000
```

## autolog period

Данная команда устанавливает период отправки логов с точки доступа на TFTP-сервер при включенной опции autolog.

Использование отрицательной формы команды (no) сбрасывает параметр на значение по умолчанию.

## Синтаксис

```
period <TIME>  
no period <TIME>
```

## Параметры

<TIME> – интервал времени в секундах, через который будет происходить автоматическая выгрузка логов, принимает значения [1.. 86400].

## Значение по умолчанию

600

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE

## Пример

```
wlc(config-wlc-ap-profile)# period 3600
```

## autolog retry

Данная команда устанавливает количество повторов отправки логов с точки доступа на TFTP-сервер при включенной опции autolog.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
retry <COUNT>  
no retry
```

## Параметры

<COUNT> – количество повторов отправки логов, допустимые значения [0..5].



**Значение по умолчанию**

3

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE

**Пример**

```
wlc(config-wlc-ap-profile)# retry 5
```

**syslog disable**

Данная команда отключает syslog на точке доступа.

Использование отрицательной формы команды (no) включает syslog.

**Синтаксис**

```
[no] disable
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE

**Пример**

```
wlc(config-wlc-ap-profile)# disable
```

**syslog mode**

Данная команда устанавливает режим работы сервиса syslog на точках доступа.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
mode { File | ServerAndFile }
no mode
```

**Параметры**

File – запись syslog будет производиться только в локальный файл на точке доступа в директории /var/log/;

ServerAndFile – syslog будет записываться в файл на точке доступа и отправляться на заданный syslog-сервер.

**Значение по умолчанию**

File

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE

**Пример**

```
wlc(config-wlc-ap-profile)# mode ServerAndFile
```

**syslog port**

Данная команда указывает номер порта сервера для отправки syslog.

Использование отрицательной формы команды (no) сбрасывает параметр на значение по умолчанию.

**Синтаксис**

```
port <NUMBER>
no port
```

**Параметры**

<NUMBER> – номер порта, принимает значения от 1 до 65535.

**Значение по умолчанию**

514

**Необходимый уровень привилегий**

10

## Командный режим

CONFIG-WLC-AP-PROFILE

### Пример

```
wlc(config-wlc-ap-profile)# port 5555
```

### syslog server

Данной командой задается адрес syslog-сервера для отправки лога с точки доступа.

Использование отрицательной формы команды (no) удаляет адрес сервера.

### Синтаксис

```
server <ADDR>  
no server
```

### Параметры

<ADDR> – IP-адрес или доменное имя сервера syslog.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE

### Пример

```
wlc(config-wlc-ap-profile)# server 192.168.1.1
```

### syslog size

Данная команда устанавливает размер файла syslog на точке доступа, при превышении которого лог будет перезаписываться, чтобы обеспечить ротацию.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
size <LIMIT>  
no size
```

## Параметры

<LIMIT> – максимальный размер файла syslog на точке доступа, принимает значение в диапазоне [0..1000] КБ.

## Значение по умолчанию

1000

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE

## Пример

```
wlc(config-wlc-ap-profile)# size 500
```

## trace

Данная команда осуществляет переход к настройке трассировок точек доступа.

Использование отрицательной формы команды (no) удаляет сконфигурированные настройки.

## Синтаксис

[no] trace

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE

## Пример

```
wlc(config-wlc-ap-profile)# trace
```

### **acsd debug-5g**

Данная команда включает дополнительное логирование для модуля 5 GHz подсистемы автовыбора радиочастотного канала на точке доступа.

Использование отрицательной формы команды (no) отключает дополнительное логирование.

#### **Синтаксис**

```
[no] acsd debug-5g
```

#### **Параметры**

Команда не содержит параметров.

#### **Значение по умолчанию**

Выключено.

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

#### **Пример**

```
wlc(config-wlc-ap-profile-trace)# acsd debug-5g
```

### **acsd debug-chanim**

Данная команда включает дополнительное логирование для модуля chanim подсистемы автовыбора радиочастотного канала на точке доступа.

Использование отрицательной формы команды (no) отключает дополнительное логирование.

#### **Синтаксис**

```
[no] acsd debug-chanim
```

#### **Параметры**

Команда не содержит параметров.

#### **Значение по умолчанию**

Выключено.

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

## Пример

```
wlc(config-wlc-ap-profile-trace)# acsd debug-chanim
```

## acsd debug-dfsр

Данная команда включает дополнительное логирование для модуля DFSr подсистемы автовыбора радиочастотного канала на точке доступа.

Использование отрицательной формы команды (no) отключает дополнительное логирование.

## Синтаксис

```
[no] acsd debug-dfsр
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Выключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

## Пример

```
wlc(config-wlc-ap-profile-trace)# acsd debug-dfsр
```

## acsd enable

Данная команда включает логирование подсистемы автовыбора радиочастотного канала на точке доступа.

Использование отрицательной формы команды (no) отключает логирование.

## Синтаксис

```
[no] acsd enable
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Выключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

## Пример

```
wlc(config-wlc-ap-profile-trace)# acsd enable
```

## acsd logfile

Данная команда задает название файла лога подсистемы автовыбора радиочастотного канала на точке доступа. Файл будет располагаться в директории /var/log/.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
acsd logfile <FILENAME>  
no acsd logfile
```

## Параметры

<FILENAME> – название файла лога в директории /var/log/ на точке доступа.

## Значение по умолчанию

acsd.log

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

## Пример

```
wlc(config-wlc-ap-profile-trace)# acsd logfile acsd.log
```

## acsd logfile-limit

Данная команда устанавливает размер файла лога подсистемы автовыбора радиочастотного канала на точке доступа, при превышении которого лог будет перезаписываться, чтобы обеспечить ротацию.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
acsd logfile-limit <LIMIT>  
no acsd logfile-limit
```

## Параметры

<LIMIT> – максимальный размер файла лога на точке доступа, принимает значение в диапазоне [1..30000] КБ.

## Значение по умолчанию

1000

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

## Пример

```
wlc(config-wlc-ap-profile-trace)# acsd logfile-limit 2000
```

## acsd loglevel

Данная команда устанавливает уровень логирования для подсистемы автовыбора радиочастотного канала на точке доступа.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
acsd loglevel { error | warn | info | debug }  
no acsd loglevel
```

## Параметры

error – лог будет записываться с уровнем error;

warn – лог будет записываться с уровнем warning;

info – лог будет записываться с уровнем info;

debug – лог будет записываться с уровнем debug.

## Значение по умолчанию

debug

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE-TRACE



## Пример

```
wlc(config-wlc-ap-profile-trace)# acsd loglevel error
```

## no acsd all

Данная команда устанавливает значения по умолчанию для всех параметров логирования подсистемы автовыбора радиочастотного канала на точке доступа.

## Синтаксис

```
no acsd all
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

## Пример

```
wlc(config-wlc-ap-profile-trace)# no acsd all
```

## airtune enable

Данная команда включает логирование агента централизованного радиопланирования на точке доступа.

Использование отрицательной формы команды (no) отключает логирование.

## Синтаксис

```
[no] airtune enable
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Выключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

### Пример

```
wlc(config-wlc-ap-profile-trace)# airtune enable
```

### airtune logfile

Данная команда задает название файла лога агента централизованного радиопланирования на точке доступа. Файл будет располагаться в директории /var/log/.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
airtune logfile <FILENAME>  
no airtune logfile
```

### Параметры

<FILENAME> – название файла лога в директории /var/log/ на точке доступа.

### Значение по умолчанию

airtune.log

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

### Пример

```
wlc(config-wlc-ap-profile-trace)# airtune logfile airtune.log
```

### airtune logfile-limit

Данная команда устанавливает размер файла лога агента централизованного радиопланирования на точке доступа, при превышении которого лог будет перезаписываться, чтобы обеспечить ротацию.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
airtune logfile-limit <LIMIT>  
no airtune logfile-limit
```

### Параметры

<LIMIT> – максимальный размер файла лога на точке доступа, принимает значение в диапазоне [1..30000] КБ.

**Значение по умолчанию**

1000

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# airtune logfile-limit 2000
```

**airtune loglevel**

Данная команда устанавливает уровень логирования агента централизованного радиопланирования на точке доступа.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
airtune loglevel { error | warn | info | debug }  
no airtune loglevel
```

**Параметры**

error – лог будет записываться с уровнем error;

warn – лог будет записываться с уровнем warning;

info – лог будет записываться с уровнем info;

debug – лог будет записываться с уровнем debug.

**Значение по умолчанию**

debug

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# airtune loglevel error
```

**no airtune all**

Данная команда устанавливает значения по умолчанию для всех параметров логирования агента централизованного радиопланирования на точке доступа.

**Синтаксис**

```
no airtune all
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# no airtune all
```

**bandsteer enable**

Данная команда включает логирование подсистемы band steer, определяющей для клиента Wi-Fi приоритетным диапазон 5 ГГц при подключении к сети на точке доступа.

Использование отрицательной формы команды (no) отключает логирование.

**Синтаксис**

```
[no] bandsteer enable
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Выключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

## Пример

```
wlc(config-wlc-ap-profile-trace)# bandsteer enable
```

## bandsteer logfile

Данная команда задает название файла лога подсистемы band steer на точке доступа. Файл будет располагаться в директории /var/log/.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
bandsteer logfile <FILENAME>
no bandsteer logfile
```

## Параметры

<FILENAME> – название файла лога в директории /var/log/ на точке доступа.

## Значение по умолчанию

bandsteerd.log

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

## Пример

```
wlc(config-wlc-ap-profile-trace)# bandsteer logfile bandsteerd.log
```

## bandsteer logfile-limit

Данная команда устанавливает размер файла лога подсистемы band steer на точке доступа, при превышении которого лог будет перезаписываться, чтобы обеспечить ротацию.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
bandsteer logfile-limit <LIMIT>
no bandsteer logfile-limit
```

## Параметры

<LIMIT> – максимальный размер файла лога на точке доступа, принимает значение в диапазоне [1..30000] КБ.

## Значение по умолчанию

1000

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

## Пример

```
wlc(config-wlc-ap-profile-trace)# bandsteer logfile-limit 2000
```

## bandsteer loglevel

Данная команда устанавливает уровень логирования подсистемы band steer на точке доступа. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
bandsteer loglevel { error | warn | debug }  
no bandsteer loglevel
```

## Параметры

error – лог будет записываться с уровнем error;

warn – лог будет записываться с уровнем warning;

debug – лог будет записываться с уровнем debug.

## Значение по умолчанию

debug

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

## Пример

```
wlc(config-wlc-ap-profile-trace)# bandsteer loglevel error
```

## bandsteer sta-mac

Данная команда задает MAC-адрес клиентской станции Wi-Fi, для которой будет записываться лог подсистемы band steer на точке доступа.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
bandsteer sta-mac <MAC-ADDRESS>
```

```
no bandsteer sta-mac
```

### Параметры

<MAC-ADDRESS> – MAC-адрес клиентской станции Wi-Fi в формате HH:HH:HH:HH:HH:HH.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

### Пример

```
wlc(config-wlc-ap-profile-trace)# bandsteer sta-mac 00:00:00:00:00:00
```

### no bandsteer all

Данная команда устанавливает значения по умолчанию для всех параметров логирования подсистемы band steer на точке доступа.

### Синтаксис

```
no bandsteer all
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

### Пример

```
wlc(config-wlc-ap-profile-trace)# no bandsteer all
```

### captive-portal enable

Данная команда включает логирование подсистемы порталной авторизации на точке доступа. Использование отрицательной формы команды (no) отключает логирование.

**Синтаксис**

```
[no] captive-portal enable
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Выключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# captive-portal enable
```

**captive-portal logfile**

Данная команда задает название файла лога подсистемы порталной авторизации на точке доступа. Файл будет располагаться в директории /var/log/.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
captive-portal logfile <FILENAME>  
no captive-portal logfile
```

**Параметры**

<FILENAME> – название файла лога в директории /var/log/ на точке доступа.

**Значение по умолчанию**

сportald.log

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# captive-portal logfile cportal.log
```



**captive-portal logfile-limit**

Данная команда устанавливает размер файла лога подсистемы порталной авторизации на точке доступа, при превышении которого лог будет перезаписываться, чтобы обеспечить ротацию.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
captive-portal logfile-limit <LIMIT>
no captive-portal logfile-limit
```

**Параметры**

<LIMIT> – максимальный размер файла лога на точке доступа, принимает значение в диапазоне [1..30000] КБ.

**Значение по умолчанию**

1000

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# captive-portal logfile-limit 2000
```

**captive-portal loglevel**

Данная команда устанавливает уровень логирования подсистемы порталной авторизации на точке доступа.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
captive-portal loglevel { error | warn | debug }
no captive-portal loglevel
```

**Параметры**

error – лог будет записываться с уровнем error;

warn – лог будет записываться с уровнем warning;

debug – лог будет записываться с уровнем debug.

**Значение по умолчанию**

debug

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# captive-portal loglevel error
```

**no captive-portal all**

Данная команда устанавливает значения по умолчанию для всех параметров логирования подсистемы портальной авторизации на точке доступа.

**Синтаксис**

```
no captive-portal all
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# no captive-portal all
```

**captive-portal apbd enable**

Данная команда включает логирование подсистемы роуминга для портальной авторизации на точке доступа.

Использование отрицательной формы команды (no) отключает логирование.

**Синтаксис**

```
[no] captive-portal apbd enable
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Выключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# captive-portal apbd enable
```

**captive-portal apbd logfile**

Данная команда задает название файла лога подсистемы роуминга для портальной авторизации на точке доступа. Файл будет располагаться в директории /var/log/.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
captive-portal apbd logfile <FILENAME>  
no captive-portal apbd logfile
```

**Параметры**

<FILENAME> – название файла лога в директории /var/log/ на точке доступа.

**Значение по умолчанию**

apbd.log

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# captive-portal apbd logfile apbd.log
```

**captive-portal apbd logfile-limit**

Данная команда устанавливает размер файла лога подсистемы роуминга для портальной авторизации на точке доступа, при превышении которого лог будет перезаписываться, чтобы обеспечить ротацию.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
captive-portal apbd logfile-limit <LIMIT>
no captive-portal apbd logfile-limit
```

**Параметры**

<LIMIT> – максимальный размер файла лога на точке доступа, принимает значение в диапазоне [1..30000] КБ.

**Значение по умолчанию**

1000

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# captive-portal apbd logfile-limit 2000
```

**captive-portal apbd loglevel**

Данная команда устанавливает уровень логирования подсистемы роуминга для порталной авторизации на точке доступа.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
captive-portal apbd loglevel { error | warn | info | debug }
no captive-portal apbd loglevel
```

**Параметры**

error – лог будет записываться с уровнем error;

warn – лог будет записываться с уровнем warning;

info – лог будет записываться с уровнем info;

debug – лог будет записываться с уровнем debug.

**Значение по умолчанию**

debug

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

## Пример

```
wlc(config-wlc-ap-profile-trace)# captive-portal apbd loglevel error
```

## no captive-portal apbd all

Данная команда устанавливает значения по умолчанию для всех параметров логирования подсистемы роуминга для портальной авторизации на точке доступа.

## Синтаксис

```
no captive-portal apbd all
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

## Пример

```
wlc(config-wlc-ap-profile-trace)# no captive-portal apbd all
```

## captive-portal redirector-debug

Данная команда включает логирование модуля перенаправления запросов HTTP на точке доступа. Использование отрицательной формы команды (no) отключает логирование.

## Синтаксис

```
[no] captive-portal redirector-debug
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Выключено.

## Необходимый уровень привилегий

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# captive-portal redirector-debug
```

**captive-portal tinyproxy enable**

Данная команда включает логирование вспомогательного HTTP проху-сервера на точке доступа. Использование отрицательной формы команды (no) отключает логирование.

**Синтаксис**

```
[no] captive-portal tinyproxy enable
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Выключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# captive-portal tinyproxy enable
```

**captive-portal tinyproxy logfile**

Данная команда задает название файла лога вспомогательного HTTP проху-сервера на точке доступа. Файл будет располагаться в директории /var/log/.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
captive-portal tinyproxy logfile <FILENAME>
no captive-portal tinyproxy logfile
```

**Параметры**

<FILENAME> – название файла лога в директории /var/log/ на точке доступа.

**Значение по умолчанию**

tinyproxy.log

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# captive-portal tinyproxy logfile tinyproxy.log
```

**captive-portal tinyproxy logfile-limit**

Данная команда устанавливает размер файла лога вспомогательного HTTP проxy-сервера на точке доступа, при превышении которого лог будет перезаписываться, чтобы обеспечить ротацию.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
captive-portal tinyproxy logfile-limit <LIMIT>  
no captive-portal tinyproxy logfile-limit
```

**Параметры**

<LIMIT> – максимальный размер файла лога на точке доступа, принимает значение в диапазоне [1..30000] КБ.

**Значение по умолчанию**

1000

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# captive-portal tinyproxy logfile-limit 2000
```

**no captive-portal tinyproxy all**

Данная команда устанавливает значения по умолчанию для всех параметров логирования вспомогательного HTTP проxy-сервера на точке доступа.

**Синтаксис**

```
no captive-portal tinyproxy all
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# no captive-portal tinyproxy all
```

**configd enable**

Данная команда включает логирование менеджера конфигурации на точке доступа. Использование отрицательной формы команды (no) отключает логирование.

**Синтаксис**

```
[no] configd enable
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Выключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# configd enable
```



## configd logfile

Данная команда задает название файла лога менеджера конфигурации на точке доступа. Файл будет располагаться в директории /var/log/.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
configd logfile <FILENAME>  
no configd logfile
```

### Параметры

<FILENAME> – название файла лога в директории /var/log/ на точке доступа.

### Значение по умолчанию

configd.log

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

### Пример

```
wlc(config-wlc-ap-profile-trace)# configd logfile configd.log
```

## configd logfile-limit

Данная команда устанавливает размер файла лога менеджера конфигурации на точке доступа, при превышении которого лог будет перезаписываться, чтобы обеспечить ротацию.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
configd logfile-limit <LIMIT>  
no configd logfile-limit
```

### Параметры

<LIMIT> – максимальный размер файла лога на точке доступа, принимает значение в диапазоне [1..30000] КБ.

### Значение по умолчанию

1000

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

### Пример

```
wlc(config-wlc-ap-profile-trace)# configd logfile-limit 2000
```

### no configd all

Данная команда устанавливает значения по умолчанию для всех параметров логирования менеджера конфигурации на точке доступа.

### Синтаксис

```
no configd all
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

### Пример

```
wlc(config-wlc-ap-profile-trace)# no configd all
```

### dmesg enable

Данная команда включает запись лога ядра в файл на точке доступа.

Использование отрицательной формы команды (no) отключает запись лога в файл.

### Синтаксис

```
[no] dmesg enable
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Выключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# dmesg enable
```

**dmesg logfile**

Данная команда задает название файла лога ядра на точке доступа. Файл будет располагаться в директории /var/log/.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
dmesg logfile <FILENAME>  
no dmesg logfile
```

**Параметры**

<FILENAME> – название файла лога в директории /var/log/ на точке доступа.

**Значение по умолчанию**

dmesg.log

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# dmesg logfile dmesg.log
```

**dmesg logfile-limit**

Данная команда устанавливает размер файла лога ядра на точке доступа, при превышении которого лог будет перезаписываться, чтобы обеспечить ротацию.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
dmesg logfile-limit <LIMIT>  
no dmesg logfile-limit
```

## Параметры

<LIMIT> – максимальный размер файла лога на точке доступа, принимает значение в диапазоне [1..30000] КБ.

## Значение по умолчанию

1000

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

## Пример

```
wlc(config-wlc-ap-profile-trace)# dmesg logfile-limit 2000
```

## no dmesg all

Данная команда устанавливает значения по умолчанию для всех параметров записи лога ядра в файл на точке доступа.

## Синтаксис

```
no dmesg all
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

## Пример

```
wlc(config-wlc-ap-profile-trace)# no dmesg all
```

## ftd enable

Данная команда включает логирование подсистемы роуминга 802.11r на точке доступа. Использование отрицательной формы команды (no) отключает логирование.

**Синтаксис**

```
[no] ftd enable
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Выключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# ftd enable
```

**ftd logfile**

Данная команда задает название файла лога подсистемы роуминга 802.11g на точке доступа. Файл будет располагаться в директории /var/log/.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
ftd logfile <FILENAME>  
no ftd logfile
```

**Параметры**

<FILENAME> – название файла лога в директории /var/log/ на точке доступа.

**Значение по умолчанию**

ftd.log

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# ftd logfile ftd.log
```

## ftd logfile-limit

Данная команда устанавливает размер файла лога подсистемы роуминга 802.11g на точке доступа, при превышении которого лог будет перезаписываться, чтобы обеспечить ротацию.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ftd logfile-limit <LIMIT>  
no ftd logfile-limit
```

### Параметры

<LIMIT> – максимальный размер файла лога на точке доступа, принимает значение в диапазоне [1..30000] КБ.

### Значение по умолчанию

1000

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

### Пример

```
wlc(config-wlc-ap-profile-trace)# ftd logfile-limit 2000
```

## no ftd all

Данная команда устанавливает значения по умолчанию для всех параметров логирования подсистемы роуминга 802.11g на точке доступа.

### Синтаксис

```
no ftd all
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

## Пример

```
wlc(config-wlc-ap-profile-trace)# no ftd all
```

## hostapd enable

Данная команда включает логирование подсистемы управления радиointерфейсами на точке доступа. Использование отрицательной формы команды (no) отключает логирование.

## Синтаксис

```
[no] hostapd enable
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Выключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

## Пример

```
wlc(config-wlc-ap-profile-trace)# hostapd enable
```

## hostapd logfile

Данная команда задает название файла лога подсистемы управления радиointерфейсами на точке доступа. Файл будет располагаться в директории /var/log/.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
hostapd logfile <FILENAME>  
no hostapd logfile
```

## Параметры

<FILENAME> – название файла лога в директории /var/log/ на точке доступа.

## Значение по умолчанию

hostapd.log

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# hostapd logfile hostapd.log
```

**hostapd logfile-limit**

Данная команда устанавливает размер файла лога подсистемы управления радиоинтерфейсами на точке доступа, при превышении которого лог будет перезаписываться, чтобы обеспечить ротацию.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
hostapd logfile-limit <LIMIT>
no hostapd logfile-limit
```

**Параметры**

<LIMIT> – максимальный размер файла лога на точке доступа, принимает значение в диапазоне [1..30000] КБ.

**Значение по умолчанию**

1000

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# hostapd logfile-limit 2000
```

**hostapd loglevel**

Данная команда устанавливает уровень логирования для подсистемы управления радиоинтерфейсами на точке доступа.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
hostapd loglevel { error | warn | info | debug | msgdump | excessive }
no hostapd loglevel
```



### Параметры

error – лог будет записываться с уровнем error;  
warn – лог будет записываться с уровнем warning;  
info – лог будет записываться с уровнем info;  
debug – лог будет записываться с уровнем debug;  
msgdump – лог будет записываться с уровнем msgdump;  
excessive – лог будет записываться с уровнем excessive;

### Значение по умолчанию

debug

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

### Пример

```
wlc(config-wlc-ap-profile-trace)# hostapd loglevel error
```

### no hostapd all

Данная команда устанавливает значения по умолчанию для всех параметров логирования подсистемы управления радиоинтерфейсами на точке доступа.

### Синтаксис

```
no hostapd all
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

### Пример

```
wlc(config-wlc-ap-profile-trace)# no hostapd all
```

### **monitord enable**

Данная команда включает логирование подсистемы мониторинга на точке доступа. Использование отрицательной формы команды (no) отключает логирование.

#### **Синтаксис**

```
[no] monitord enable
```

#### **Параметры**

Команда не содержит параметров.

#### **Значение по умолчанию**

Выключено.

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

#### **Пример**

```
wlc(config-wlc-ap-profile-trace)# monitord enable
```

### **monitord logfile**

Данная команда задает название файла лога подсистемы мониторинга на точке доступа. Файл будет располагаться в директории /var/log/.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### **Синтаксис**

```
monitord logfile <FILENAME>  
no monitord logfile
```

#### **Параметры**

<FILENAME> – название файла лога в директории /var/log/ на точке доступа.

#### **Значение по умолчанию**

monitord.log

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

## Пример

```
wlc(config-wlc-ap-profile-trace)# monitord logfile monitord.log
```

## monitord logfile-limit

Данная команда устанавливает размер файла лога подсистемы мониторинга на точке доступа, при превышении которого лог будет перезаписываться, чтобы обеспечить ротацию.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
monitord logfile-limit <LIMIT>
no monitord logfile-limit
```

### Параметры

<LIMIT> – максимальный размер файла лога на точке доступа, принимает значение в диапазоне [1..30000] КБ.

### Значение по умолчанию

1000

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

## Пример

```
wlc(config-wlc-ap-profile-trace)# monitord logfile-limit 2000
```

## monitord loglevel

Данная команда устанавливает уровень логирования для подсистемы мониторинга на точке доступа.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
monitord loglevel { error | warn | info | debug }
no monitord loglevel
```

### Параметры

error – лог будет записываться с уровнем error;

warn – лог будет записываться с уровнем warning;

info – лог будет записываться с уровнем info;

debug – лог будет записываться с уровнем debug.

**Значение по умолчанию**

debug

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# monitord loglevel error
```

**no monitord all**

Данная команда устанавливает значения по умолчанию для всех параметров логирования подсистемы мониторинга на точке доступа.

**Синтаксис**

```
no monitord all
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# no monitord all
```

**netconf enable**

Данная команда включает логирование агента NETCONF на точке доступа.  
Использование отрицательной формы команды (no) отключает логирование.

**Синтаксис**

```
[no] netconf enable
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Выключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

## Пример

```
wlc(config-wlc-ap-profile-trace)# netconf enable
```

## netconf logfile

Данная команда задает название файла лога агента NETCONF на точке доступа. Файл будет располагаться в директории /var/log/.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
netconf logfile <FILENAME>  
no netconf logfile
```

## Параметры

<FILENAME> – название файла лога в директории /var/log/ на точке доступа.

## Значение по умолчанию

netconf.log

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

## Пример

```
wlc(config-wlc-ap-profile-trace)# netconf logfile netconf.log
```

## netconf logfile-limit

Данная команда устанавливает размер файла лога агента NETCONF на точке доступа, при превышении которого лог будет перезаписываться, чтобы обеспечить ротацию.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
netconf logfile-limit <LIMIT>  
no netconf logfile-limit
```

### Параметры

<LIMIT> – максимальный размер файла лога на точке доступа, принимает значение в диапазоне [1..30000] КБ.

### Значение по умолчанию

1000

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

### Пример

```
wlc(config-wlc-ap-profile-trace)# netconf logfile-limit 2000
```

## no netconf all

Данная команда устанавливает значения по умолчанию для всех параметров логирования агента NETCONF на точке доступа.

### Синтаксис

```
no netconf all
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

## Пример

```
wlc(config-wlc-ap-profile-trace)# no netconf all
```

## networkd enable

Данная команда включает логирование системного менеджера на точке доступа. Использование отрицательной формы команды (no) отключает логирование.

## Синтаксис

```
[no] networkd enable
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Выключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

## Пример

```
wlc(config-wlc-ap-profile-trace)# networkd enable
```

## networkd logfile

Данная команда задает название файла лога системного менеджера на точке доступа. Файл будет располагаться в директории /var/log/.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
networkd logfile <FILENAME>  
no networkd logfile
```

## Параметры

<FILENAME> – название файла лога в директории /var/log/ на точке доступа.

## Значение по умолчанию

networkd.log

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

## Пример

```
wlc(config-wlc-ap-profile-trace)# networkd logfile networkd.log
```

## networkd logfile-limit

Данная команда устанавливает размер файла лога системного менеджера на точке доступа, при превышении которого лог будет перезаписываться, чтобы обеспечить ротацию.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
networkd logfile-limit <LIMIT>  
no networkd logfile-limit
```

## Параметры

<LIMIT> – максимальный размер файла лога на точке доступа, принимает значение в диапазоне [1..30000] КБ.

## Значение по умолчанию

1000

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

## Пример

```
wlc(config-wlc-ap-profile-trace)# networkd logfile-limit 2000
```

## networkd loglevel

Данная команда устанавливает уровень логирования системного менеджера на точке доступа.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
networkd loglevel { error | warn | info | debug }  
no networkd loglevel
```



**Параметры**

error – лог будет записываться с уровнем error;  
warn – лог будет записываться с уровнем warning;  
info – лог будет записываться с уровнем info;  
debug – лог будет записываться с уровнем debug.

**Значение по умолчанию**

debug

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# networkd loglevel error
```

**no networkd all**

Данная команда устанавливает значения по умолчанию для всех параметров логирования системного менеджера на точке доступа.

**Синтаксис**

```
no networkd all
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# no networkd all
```

## **snmp enable**

Данная команда включает логирование агента SNMP на точке доступа.

Использование отрицательной формы команды (no) отключает логирование.

### **Синтаксис**

```
[no] snmp enable
```

### **Параметры**

Команда не содержит параметров.

### **Значение по умолчанию**

Выключено.

### **Необходимый уровень привилегий**

10

### **Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

### **Пример**

```
wlc(config-wlc-ap-profile-trace)# snmp enable
```

## **snmp logfile**

Данная команда задает название файла лога агента SNMP на точке доступа. Файл будет располагаться в директории /var/log/.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### **Синтаксис**

```
snmp logfile <FILENAME>  
no snmp logfile
```

### **Параметры**

<FILENAME> – название файла лога в директории /var/log/ на точке доступа.

### **Значение по умолчанию**

snmp.log

### **Необходимый уровень привилегий**

10

### **Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

## Пример

```
wlc(config-wlc-ap-profile-trace)# snmp logfile snmp.log
```

## snmp logfile-limit

Данная команда устанавливает размер файла лога агента SNMP на точке доступа, при превышении которого лог будет перезаписываться, чтобы обеспечить ротацию.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
snmp logfile-limit <LIMIT>  
no snmp logfile-limit
```

## Параметры

<LIMIT> – максимальный размер файла лога на точке доступа, принимает значение в диапазоне [1..30000] КБ.

## Значение по умолчанию

1000

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

## Пример

```
wlc(config-wlc-ap-profile-trace)# snmp logfile-limit 2000
```

## no snmp all

Данная команда устанавливает значения по умолчанию для всех параметров логирования агента SNMP на точке доступа.

## Синтаксис

```
no snmp all
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# no snmp all
```

**wlc-activator enable**

Данная команда включает логирование на точке доступа для сервиса инициализации точек доступа. Использование отрицательной формы команды (no) отключает логирование.

**Синтаксис**

```
[no] wlc-activator enable
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Выключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# wlc-activator enable
```

**wlc-activator logfile**

Данная команда задает название файла лога сервиса инициализации точек доступа. Файл будет располагаться в директории /var/log/.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
wlc-activator logfile <FILENAME>  
no wlc-activator logfile
```

**Параметры**

<FILENAME> – название файла лога в директории /var/log/ на точке доступа.

**Значение по умолчанию**

service-activator-wlc.log

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# wlc-activator logfile service-activator-wlc.log
```

**wlc-activator logfile-limit**

Данная команда устанавливает размер файла лога сервиса инициализации точек доступа, при превышении которого лог будет перезаписываться, чтобы обеспечить ротацию.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
wlc-activator logfile-limit <LIMIT>  
no wlc-activator logfile-limit
```

**Параметры**

<LIMIT> – максимальный размер файла лога на точке доступа, принимает значение в диапазоне [1..30000] КБ.

**Значение по умолчанию**

1000

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# wlc-activator logfile-limit 2000
```

**no wlc-activator all**

Данная команда устанавливает значения по умолчанию для всех параметров логирования сервиса и вспомогательного сервера HTTPS для инициализации точек доступа.

**Синтаксис**

```
no wlc-activator all
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# no wlc-activator all
```

**wlc-activator server enable**

Данная команда включает логирование на точке доступа вспомогательного сервера HTTPS для инициализации точек доступа.

Использование отрицательной формы команды (no) отключает логирование.

**Синтаксис**

```
[no] wlc-activator server enable
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Выключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

## Пример

```
wlc(config-wlc-ap-profile-trace)# wlc-activator server enable
```

## wlc-activator server logfile

Данная команда задает название файла лога вспомогательного сервера HTTPS для инициализации точек доступа. Файл будет располагаться в директории /var/log/.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
wlc-activator server logfile <FILENAME>
no wlc-activator server logfile
```

## Параметры

<FILENAME> – название файла лога в директории /var/log/ на точке доступа.

## Значение по умолчанию

service-activator-server-wlc.log

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

## Пример

```
wlc(config-wlc-ap-profile-trace)# wlc-activator server logfile service-activator-server-wlc.log
```

## wlc-activator server logfile-limit

Данная команда устанавливает размер файла лога вспомогательного сервера HTTPS для инициализации точек доступа, при превышении которого лог будет перезаписываться, чтобы обеспечить ротацию.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
wlc-activator server logfile-limit <LIMIT>
no wlc-activator server logfile-limit
```

## Параметры

<LIMIT> – максимальный размер файла лога на точке доступа, принимает значение в диапазоне [1..30000] КБ.

**Значение по умолчанию**

1000

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# wlc-activator server logfile-limit 2000
```

**no wlc-activator server all**

Данная команда устанавливает значения по умолчанию для всех параметров логирования вспомогательного сервера HTTPS для инициализации точек доступа.

**Синтаксис**

```
no wlc-activator server all
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# no wlc-activator server all
```

**wids enable**

Данная команда включает логирование на точке доступа сервиса WIDS.

Использование отрицательной формы команды (no) отключает логирование.

**Синтаксис**

```
[no] wids enable
```



## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Выключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

## Пример

```
wlc(config-wlc-ap-profile-trace)# wids enable
```

## wids logfile

Данная команда задает название файла лога сервиса WIDS. Файл будет располагаться в директории /var/log/.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
wids logfile <FILENAME>  
no wids logfile
```

## Параметры

<FILENAME> – название файла лога в директории /var/log/ на точке доступа.

## Значение по умолчанию

widsd.log

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WLC-AP-PROFILE-TRACE

## Пример

```
wlc(config-wlc-ap-profile-trace)# wids logfile widsd.log
```

**wids logfile-limit**

Данная команда устанавливает размер файла лога сервиса WIDS на точке доступа, при превышении которого лог будет перезаписываться, чтобы обеспечить ротацию.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
wids logfile-limit <LIMIT>
no wids logfile-limit
```

**Параметры**

<LIMIT> – максимальный размер файла лога на точке доступа, принимает значение в диапазоне [1..30000] КБ.

**Значение по умолчанию**

1000

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# wids logfile-limit 2000
```

**wids loglevel**

Данная команда устанавливает уровень логирования сервиса WIDS на точке доступа.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
wids loglevel { error | warn | info | debug | dump }
no wids loglevel
```

**Параметры**

error – лог будет записываться с уровнем error;

warn – лог будет записываться с уровнем warning;

info – лог будет записываться с уровнем info;

debug – лог будет записываться с уровнем debug;

dump - лог будет записываться с уровнем dump.

**Значение по умолчанию**

debug

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# wids loglevel error
```

**no wids all**

Данная команда устанавливает значения по умолчанию для всех параметров логирования сервиса WIDS точки доступа.

**Синтаксис**

```
no wids all
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-AP-PROFILE-TRACE

**Пример**

```
wlc(config-wlc-ap-profile-trace)# no wids all
```

**Настройка журналирования событий WLC****wlc-journal**

Данная команда осуществляет переход к настройке ограничения максимального размера для всех журналов либо конкретного журнала: событий по ТД, событий по клиентам или событий WIDS.

Использование отрицательной формы команды (no) удаляет заданные ограничения.

**Синтаксис**

```
wlc-journal { all | ap | clients | wids }
no wlc-journal { all | ap | clients | wids }
```

**Параметры**

all – установка ограничения для всех журналов;  
 ap – установка ограничения для журнала событий по точкам доступа;  
 clients – установка ограничения для журнала событий по клиентам;  
 wids – установка ограничения для журнала событий WIDS.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
wlc(config)# wlc-journal ap
```

**limit**

Данная команда позволяет установить ограничение записи событий в журналах.  
 Использование отрицательной формы команды (no) удаляет заданное ограничение.

**Синтаксис**

```
limit { <DAYS> }  
no limit
```

**Параметры**

<DAYS> – параметр определяет сколько дней в журнале будет сохраняться, принимает значения в диапазоне [0..2000]

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WLC-JOURNAL-ALL  
 CONFIG-WLC-JOURNAL-AP  
 CONFIG-WLC-JOURNAL-CLIENTS

## CONFIG-WLC-JOURNAL-WIDS

**Пример**

```
wlc(config-wlc-journal-all)# limit days 4
```

**Настройка RADIUS-сервера**

- [acct-port](#)
- [auth-port](#)
- [domain](#)
- [crypto ca](#)
- [crypto cert](#)
- [crypto private-key](#)
- [enable](#)
- [host](#)
- [key](#)
- [nas](#)
- [nas-ip-address](#)
- [network](#)
- [password](#)
- [port](#)
- [priority](#)
- [proxy-mode](#)
- [radius-server local](#)
- [server-type](#)
- [tls mode](#)
- [upstream-server](#)
- [user](#)
- [virtual-server](#)
- [vlan](#)

**acct-port**

Данной командой задается номер порта для обмена данными с удаленным RADIUS-сервером при выполнении аккаунтинга.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
acct-port <PORT>  
no acct-port
```

**Параметры**

<PORT> – номер UDP-порта для обмена данными с удаленным сервером, принимает значения [1..65535].

**Значение по умолчанию**

1813

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-RADIUS-VSERVER

**Пример**

```
wlc-30(config-radius-vserver)# acct-port 1813
```

**auth-port**

Данной командой задаётся номер порта для обмена данными с удаленным RADIUS-сервером при выполнении аутентификации и авторизации.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
auth-port <PORT>
```

```
no auth-port
```

**Параметры**

<PORT> – номер UDP-порта для обмена данными с удаленным сервером, принимает значения [1..65535].

**Значение по умолчанию**

1812

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-RADIUS-VSERVER

**Пример:**

```
wlc-30(config-radius-vserver)# auth-port 1812
```

**domain**

Данная команда создает/назначает домен.

Использование отрицательной формы команды (no) удаляет домен.

## Синтаксис

```
domain <DOMAIN>  
no domain { <DOMAIN> | all }
```

## Параметры

<DOMAIN> – идентификатор домена, задается строкой до 235 символов;  
all – команда удаляет все созданные домены.

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-RADIUS

## Пример

```
wlc-30(config-radius)# domain test
```

## crypto ca

Данная команда указывает, какой сертификат использовать в качестве ca.

Использование отрицательной формы команды (no) устанавливает ca-сертификат по умолчанию.

## Синтаксис

```
crypto ca <NAME>  
no crypto ca
```

## Параметры

<NAME> – название сертификата, загруженного по пути crypto:cert, задается строкой до 31 символа.

## Значение по умолчанию

default\_ca.pem

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-RADIUS

## Пример

```
wlc-30(config-radius)# crypto ca default_ca.pem
```

## crypto cert

Данная команда указывает, какой сертификат использовать.

Использование отрицательной формы команды (no) устанавливает сертификат по умолчанию.

## Синтаксис

```
crypto cert <NAME>  
no crypto cert
```

## Параметры

<NAME> – название сертификата, загруженного по пути crypto:cert или сгенерированного командой [crypto generate cert](#), задается строкой до 31 символа.

## Значение по умолчанию

default\_cert.pem

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-RADIUS

CONFIG-RADIUS-USER

## Пример

```
wlc-30(config-radius)# crypto cert default_cert.pem
```

## crypto private-key

Данная команда указывает какой приватный ключ использовать для сертификата сервера.

Использование отрицательной формы команды (no) устанавливает ключ по умолчанию.

## Синтаксис

```
crypto private-key <NAME>  
no crypto private-key
```

## Параметры

<NAME> – название сертификата, загруженного по пути crypto:private-key, задается строкой до 31 символа.



**Значение по умолчанию**

default\_cert\_key.pem

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-RADIUS

**Пример**

```
wlc-30(config-radius)# crypto private-key default_cert_key.pem
```

**enable**

Данная команда предназначена для активации функций сервера.

Использование отрицательной формы команды (no) отключает функции.

**Синтаксис**

[no] enable

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-RADIUS

CONFIG-RADIUS-VSERVER

**Пример**

```
wlc-30(config-radius)# enable
```

**host**

Данной командой задаётся IP-адрес удаленного хоста для аутентификации и авторизации.

Использование отрицательной формы команды (no) удаляет хост.

**Синтаксис**

```
host <ADDR>
no host
```

**Параметры**

<ADDR> – IP-адрес удаленного хоста, задаётся в виде: AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-RADIUS-UPSTREAM-SERVER

**Пример:**

```
wlc-30(config-radius-upstream-server)# host 192.168.1.1
```

**key**

Данной командой задаётся ключ аутентификации.

Использование отрицательной формы команды (no) удаляет заданный ключ.

**Синтаксис**

```
key ascii-text { <KEY> | encrypted <ENCRYPTED-KEY> }
no key
```

**Параметры**

<KEY> – строка из [4..64] ASCII-символов;

<ENCRYPTED-KEY> – зашифрованный ключ, задаётся строкой [8..128] символов.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-RADIUS-NAS

## CONFIG-RADIUS-UPSTREAM-SERVER

**Пример:**

```
wlc-30(config-radius-upstream-server)# key ascii-text password
```

**nas**

Данной командой задаётся NAS.

Использование отрицательной формы команды (no) удаляет NAS.

**Синтаксис**

```
nas <NAME>
```

```
no nas { <NAME> | all }
```

**Параметры**

<NAME> – название, задается строкой до 235 символов;

all – команда удаляет все NAS.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-RADIUS

**Пример:**

```
wlc-30(config-radius)# nas test
```

**nas-ip-address**

Данной командой задаётся IP-адрес, на который будет заменен NAS IP при проксировании пакетов RADIUS.

Использование отрицательной формы команды (no) удаляет IP-адрес, поле NAS IP будет содержать адрес точки доступа.

**Синтаксис**

```
nas-ip-address <ADDR>
```

```
no nas-ip-address
```

**Параметры**

<ADDR> – IP-адрес, задаётся в виде: AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-RADIUS-VSERVER

```
wlc-30(config-radius-vserver)# nas-ip-address 192.168.1.100
```

**network**

Данной командой задаётся доступное сетевое пространство.

Использование отрицательной формы команды (no) удаляет сеть.

**Синтаксис**

```
network <ADDR/LEN>
no network
```

**Параметры**

<ADDR/LEN> – IP-адрес и маска подсети, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-RADIUS-NAS

**Пример:**

```
wlc-30(config-radius-nas)# network 192.168.1.0/24
```

**password**

Команда для установки пароля определенному пользователю.

Использование отрицательной формы команды (no) удаляет пароль пользователя.

**Синтаксис**

```
password ascii-text { <PASSWORD> | encrypted <ENCRYPTED-PASSWORD> }
no password
```

**Параметры**

<PASSWORD> – пароль, задаётся строкой [8.. 64] символа.

<ENCRYPTED-PASSWORD> – зашифрованный пароль, задается строкой [16-128] символов.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-RADIUS-USER

**Пример**

```
wlc-30(config-radius-user)# password ascii-text password
```

**port**

Данной командой задаётся номер порта для обмена данными с удаленным сервером.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
port <PORT>
no port
```

**Параметры**

<PORT> – номер TCP/UDP-порта для обмена данными с удаленным сервером, принимает значения [1..65535].

**Значение по умолчанию**

0

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-RADIUS-UPSTREAM-SERVER

**Пример:**

```
wlc-30(config-radius-upstream-server)# port 1812
```

**priority**

Данной командой задаётся приоритет использования удаленного сервера. Чем ниже значение, тем приоритетнее сервер.

Использование отрицательной формы команды (no) удаляет параметр.

**Синтаксис**

```
priority <PRIORITY>
```

```
no priority
```

**Параметры**

<PRIORITY> – приоритет использования удаленного сервера, принимает значения [1..100].

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-RADIUS-UPSTREAM-SERVER

**Пример:**

```
wlc-30(config-radius-upstream-server)# priority 1
```

**proxy-mode**

Данной командой активируется режим проксирования RADIUS-сервера.

Использование отрицательной формы команды (no) отключает режим проксирования.

**Синтаксис**

[no] proxy-mode

**Параметры**

Отсутствуют.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-RADIUS-VSERVER

**Пример:**

```
wlc-30(config-radius-vserver)# proxy-mode
```

**radius-server local**

Данной командой осуществляется переход в настройки локального RADIUS-сервера.

Использование отрицательной формы команды (no) удаляет сконфигурированные настройки.

**Синтаксис**

[no] radius-server local

**Параметры**

Отсутствуют.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример:**

```
wlc-30(config)# radius-server local
```

**server-type**

Данная команда указывает тип сервера.

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

**Синтаксис**

```
server-type { auth | acct | auth+acct }
no server-type
```

**Параметры**

auth – RADIUS-сервер будет использоваться для аутентификации, авторизации;

acct – RADIUS-сервер будет использоваться для сбора акаунтинга;

auth+acct – RADIUS-сервер будет выполнять обе функции.

**Значение по умолчанию**

auth

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-RADIUS-UPSTREAM-SERVER

**Пример:**

```
wlc-30(config-radius-upstream-server)# server-type auth+acct
```

**tls mode**

Данная команда предназначена для активации возможности использования авторизации по сертификатам TLS.

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

**Синтаксис**

```
tls mode {domain | disable}
no tls mode
```

**Параметры**

domain – включить возможность авторизации по сертификатам TLS в домене;



disable – отключить возможность авторизации по сертификатам TLS.

### Значение по умолчанию

disable

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-RADIUS-DOMAIN

### Пример:

```
wlc-30(config-radius-domain)# tls mode domain
```

### upstream-server

Данная команда добавляет upstream RADIUS-сервер.

Использование отрицательной формы команды (no) удаляет upstream RADIUS-сервер.

### Синтаксис

```
upstream-server <NAME>  
no upstream-server { <NAME> | all }
```

### Параметры

<NAME> – название upstream RADIUS-сервера, задается строкой до 246 символов;

all – команда удаляет все upstream RADIUS-сервера.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-RADIUS-VSERVER

### Пример:

```
wlc-30(config-radius-vserver)# upstream-server test
```

**user**

Данной командой выполняется добавление пользователя в локальную базу пользователей и осуществляется переход в режим настройки параметров пользователя.

Использование отрицательной формы команды (no) удаляет конкретного пользователя или сразу всех пользователей из системы.

**Синтаксис**

```
user <NAME>
no user { <NAME> | all }
```

**Параметры**

<NAME> – имя пользователя, задается строкой до 246 символов;

all – команда удаляет всех созданных пользователей.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-RADIUS-DOMAIN

**Пример**

```
wlc-30(config-radius-domain)# user test
```

**virtual-server**

Данной командой выполняется добавление виртуального RADIUS-сервера и осуществляется переход в режим настройки его параметров.

Использование отрицательной формы команды (no) удаляет виртуальный RADIUS-сервер.

**Синтаксис**

```
virtual-server <NAME>
no virtual-server { <NAME> | all }
```

**Параметры**

<NAME> – название виртуального RADIUS-сервера, задается строкой до 235 символов;

all – команда удаляет все созданные RADIUS-сервера.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-RADIUS

**Пример**

```
wlc-30(config-radius)# virtual-server test
```

**vlan**

Данной командой выполняется добавление vlan для пользователя.

Использование отрицательной формы команды (no) удаляет vlan пользователя.

**Синтаксис**

```
vlan <ID>  
no vlan
```

**Параметры**

<ID> – номер vlan, доступны значения от 1 до 4094.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-RADIUS-USER

**Пример**

```
wlc-30(config-radius-user)# vlan 2
```

**Настройка WEB-сервера**

- [ip http port](#)
- [ip http server](#)
- [ip https crypto ca](#)

- [ip https crypto cert](#)
- [ip https crypto private-key](#)
- [ip https port](#)
- [ip https server](#)
- [ip http failover](#)

### **ip http port**

Данной командой определяется порт HTTP-сервера на контроллере.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### **Синтаксис**

```
ip http port <PORT>
```

```
no ip http port
```

#### **Параметры**

<PORT> – номер порта, указывается в диапазоне [1..65535].

#### **Значение по умолчанию**

80

#### **Необходимый уровень привилегий**

15

#### **Командный режим**

CONFIG

#### **Пример**

```
wlc(config)# ip http port 80
```

### **ip http server**

Данной командой включается HTTP-сервер на контроллере.

Использование отрицательной формы команды (no) отключает HTTP-сервер.

#### **Синтаксис**

```
[no] ip http server
```

#### **Параметры**

Отсутствуют.

#### **Значение по умолчанию**

http-сервер выключен.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
wlc(config)# no ip http server
```

**ip https crypto ca**

Данная команда указывает какой сертификат использовать в качестве ca.

Использование отрицательной формы команды (no) устанавливает ca-сертификат по умолчанию.

**Синтаксис**

```
ip https crypto ca <NAME>  
no ip https crypto ca
```

**Параметры**

<NAME> – название сертификата, загруженного по пути crypto:cert, задается строкой до 31 символа.

**Значение по умолчанию**

default\_ca.pem

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
wlc(config)# ip https crypto ca test_ca.pem
```

**ip https crypto cert**

Данная команда указывает какой сертификат использовать в качестве сертификата https-сервера.

Использование отрицательной формы команды (no) устанавливает сертификат по умолчанию.

**Синтаксис**

```
ip https crypto cert <NAME>  
no ip https crypto cert
```

## Параметры

<NAME> – название сертификата, загруженного по пути crypto:cert, задается строкой до 31 символа.

## Значение по умолчанию

default\_cert.pem

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
wlc(config)# ip https crypto cert test_cert.pem
```

## ip https crypto private-key

Данная команда указывает какой приватный ключ использовать для сертификата сервера.

Использование отрицательной формы команды (no) устанавливает ключ по умолчанию.

## Синтаксис

```
ip https crypto private-key <NAME>  
no ip https crypto private-key
```

## Параметры

<NAME> – название приватного ключа, загруженного по пути crypto:private-key, задается строкой до 31 символа.

## Значение по умолчанию

default\_cert\_key.pem

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
wlc(config)# ip https crypto private-key test_cert_key.pem
```

**ip https port**

Данной командой определяется порт http-сервера на контроллере.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
ip https port <PORT>
```

```
no ip https port
```

**Параметры**

<PORT> – номер порта, указывается в диапазоне [1..65535].

**Значение по умолчанию**

443

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
wlc(config)# ip https port 443
```

**ip https server**

Данной командой включается https-сервер на контроллере.

Использование отрицательной формы команды (no) отключает https-сервер.

**Синтаксис**

```
[no] ip https server
```

**Параметры**

Отсутствуют.

**Значение по умолчанию**

https-сервер выключен.

**Необходимый уровень привилегий**

10

## Командный режим

CONFIG

### Пример

```
wlc(config)# no ip https server
```

### ip http failover

Данная команда активирует работу резервирования пользовательских настроек в WEB. Указание адресации и группы описано в разделе «[Резервирование](#)».

Использование отрицательной формы команды (no) выключает настройку синхронизации WEB-профилей.

### Синтаксис

```
[no] ip http failover
```

### Параметры

Отсутствуют.

### Значение по умолчанию

Отключено

### Необходимый уровень привилегий

10

## Командный режим

CONFIG

### Пример

```
wlc(config)# ip http failover
```

## Настройка авторизации пользователей через LDAP

- [ldap-server host](#)
- [ldap-server bind authenticate root-dn](#)
- [ldap-server bind authenticate root-password](#)
- [aaa ldap profile](#)
- [base-dn](#)
- [ldap-server host](#)
- [ldap-mode](#)
- [ldap-profile](#)



## ldap-server host

Данная команда используется для добавления LDAP-сервера в список используемых серверов и перехода в командный режим LDAP SERVER.

Использование отрицательной формы команды (no) удаляет заданный LDAP-сервер.

### Синтаксис

```
[no] ldap-server host { <ADDR> | <IPV6-ADDR> | <LDAP-HOST-NAME> } [ vrf <VRF> ]
```

### Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа;

<ADDR> – IP-адрес LDAP-сервера, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес LDAP-сервера, задается в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<LDAP-HOST-NAME> – DNS-имя LDAP-сервера, задается строкой до 31 символа. Для корректного сопоставления DNS-имени хоста с IP-адресом на маршрутизаторе должен быть запущен и настроен функционал [domain lookup enable](#).

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

### Пример:

```
wlc(config)# ldap-server host 10.100.100.1
wlc(config-ldap-server)#
```

## ldap-server bind authenticate root-dn

Данной командой задается DN (Distinguished name) пользователя с правами администратора, под которым будет происходить авторизация на LDAP-сервере при поиске пользователей.

Использование отрицательной формы команды (no) удаляет заданный DN пользователя.

### Синтаксис

```
ldap-server bind authenticate root-dn <NAME>
```

```
no bind authenticate root-dn
```

### Параметры

<NAME> – DN пользователя с правами администратора, задается строкой до 255 символов.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример:**

```
wlc(config)# ldap-server bind authenticate root-dn "cn=admin,dc=example,dc=com"
```

**ldap-server bind authenticate root-password**

Данной командой задаётся пароль пользователя с правами администратора, под которым будет происходить авторизация на LDAP-сервере при поиске пользователей.

Использование отрицательной формы команды (no) удаляет заданный пароль пользователя.

**Синтаксис**

```
ldap-server bind authenticate root-password ascii-text { <TEXT> | encrypted <ENCRYPTED-TEXT> }
```

```
no bind authenticate root-password
```

**Параметры**

<TEXT> – строка [8..16] ASCII-символов;

<ENCRYPTED-TEXT> – зашифрованный пароль, размером [8..16] байт, задаётся строкой [16..32] символов.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример:**

```
wlc(config)# ldap-server bind authenticate root-password ascii-text 12345678
```

**aaa ldap profile**

Данная команда используется для добавления профиля LDAP-серверов и перехода в командный режим LDAP SERVER PROFILE.

Использование отрицательной формы команды (no) удаляет заданный профиль LDAP-серверов.

**Синтаксис**

```
[no] aaa radius-profile <NAME>
```

**Параметры**

<NAME> – имя профиля LDAP-серверов, задается строкой до 31 символа.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример:**

```
esr(config)# aaa ldap-profile profile1
esr(config-aaa-ldap-profile)#
```

**base-dn**

Данной командой задаётся базовый DN (Distinguished name), который будет использоваться при поиске пользователей.

Использование отрицательной формы команды (no) удаляет заданный базовый DN.

**Синтаксис**

```
base-dn <NAME>
```

```
no base-dn
```

**Параметры**

<NAME> – базовый DN, задается строкой до 255 символов.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-AAA-LDAP-PROFILE

**Пример:**

```
wlc-30(config-aaa-ldap-profile)# base-dn "ou=Devs,dc=eltex,dc=ru"
```

**ldap-server host**

Данная команда используется для добавления LDAP-сервера в профиль LDAP-серверов.

Использование отрицательной формы команды (no) удаляет заданный LDAP-сервер из профиля.

### Синтаксис

```
[no] radius-server host { <ADDR> | <IPV6-ADDR> }
```

### Параметры

<ADDR> – IP-адрес LDAP-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес LDAP-сервера, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-LDAP-SERVER-PROFILE

### Пример:

```
esr(config-aaa-ldap-profile)# ldap-server host 10.100.100.1
```

### ldap-mode

Данной командой активируется режим поддержки LDAP локального RADIUS-сервера.

Использование отрицательной формы команды (no) отключает режим LDAP.

### Синтаксис

```
[no] ldap-mode
```

### Параметры

Отсутствуют.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-RADIUS-VSERVER

**Пример:**

```
wlc-30(config-radius-vserver)# ldap-mode
```

**ldap-profile**

Данной командой выбирается один из созданных LDAP-профилей для работы в локальном RADIUS-сервере.

Использование отрицательной формы (no) отключает используемый профиль LDAP.

**Синтаксис**

```
[no] ldap-profile <NAME>
```

**Параметры**

<NAME> – имя профиля LDAP-серверов, задается строкой до 31 символа.

**Значение по умолчанию**

Отсутствует

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-RADIUS

**Пример**

```
wlc-30(config-radius)# ldap-profile tester
```

**Работа с сертификатами**

- [crypto generate cert](#)
- [crypto generate csr](#)
- [crypto generate pfx](#)
- [crypto generate private-key](#)
- [update crypto default](#)

**crypto generate cert**

Команда предназначена для генерации сертификатов. При работе в режиме контроллера WLC используется для генерации пользовательских сертификатов для авторизации по протоколу TLS.

**Синтаксис**

```
crypto generate cert csr <CSR> ca <CA> private-key <KEY> filename <NAME_CERT>
```

## Параметры

<CSR> – имя существующего csr-файла;

<CA> – имя корневого сертификата, если не был создан новый, то можно воспользоваться дефолтным, который генерируется при старте устройства (default\_ca.pem);

<KEY> – имя ключа корневого сертификата, если не был создан новый, то можно воспользоваться дефолтным, который генерируется при старте устройства (default\_ca\_key.pem);

<NAME\_CERT> – имя создаваемого сертификата, задается строкой до 31 символа, может содержать латинские буквы разного регистра, цифры и символы '.', '-', '\_', '@'.

## Необходимый уровень привилегий

10

## Командный режим

ROOT

## Пример

```
wlc# crypto generate cert csr tester.csr ca default_ca.pem private-key default_ca_key.pem
filename tester.crt
```

## crypto generate csr

Команда предназначена для генерации csr-файлов, необходимых для генерации сертификатов. При работе в режиме контроллера WLC csr используется при генерации пользовательских сертификатов для авторизации по протоколу TLS.

## Синтаксис

```
crypto generate csr private-key <USER_KEY> [alternative-name <ALT_NAME> | common-name
<COM_NAME> | country <COUNTRY> | email-address <EMAIL> | locality <LOC> | organization
<ORG_NAME> | organizational-unit <UNIT_NAME> |
state <STATE>] filename <CSR>
```

## Параметры

<USER\_KEY> – имя ключа пользовательского сертификата;

<ALT\_NAME> – альтернативное имя SAN (Subject Alternative Names), позволяющее объединять несколько ресурсов под одним именем. Задается строкой от 5 до 255 символов;

<COM\_NAME> – доменное имя, для которого заказывается сертификат, задается строкой до 64 символов. Если команда используется для генерации сертификата пользователя Wi-Fi, то нужно указать также имя пользователя в формате: user@domain;

<COUNTRY> – код страны регистрации организации, задается строкой из 2 символов;

<EMAIL> – e-mail для связи, задается строкой от 3 до 64 символов;

<LOC> – город регистрации организации, задается строкой до 128 символов;

<ORG\_NAME> – название организации, задается строкой до 64 символов;

<UNIT\_NAME> – название отдела организации, задается строкой до 64 символов;

<STATE> – область, регион регистрации организации, задается строкой до 64 символов;

<CSR> – имя создаваемого csr-файла, задается строкой до 31 символа, может содержать латинские буквы разного регистра, цифры и символы '.', '-', '\_', '@'.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример

```
wlc# crypto generate csr private-key tester.pem common-name tester@wlc.root filename tester.csr
```

### crypto generate pfx

Команда предназначена для генерации pfx-контейнера. При работе в режиме контроллера WLC контейнер используется при авторизации пользователя по протоколу TLS.

### Синтаксис

```
crypto generate pfx private-key <USER_KEY> cert <NAME_CERT> ca <CA> [password ascii-text <PFX_PASSWORD>] filename <PFX>
```

### Параметры

<USER\_KEY> – имя ключа пользовательского сертификата;

<NAME\_CERT> – имя созданного существующего сертификата пользователя;

<CA> – имя корневого сертификата, если не был создан новый, то можно воспользоваться дефолтным, который генерируется при старте устройства (default\_ca.pem);

<PFX\_PASSWORD> – пароль для открытия контейнера, задается строкой до 64 символов;

<PFX> – имя контейнера, задается строкой до 31 символа, может содержать латинские буквы разного регистра, цифры и символы '.', '-', '\_', '@'. В случае генерации контейнера для пользователя Wi-Fi нужно указывать расширение .p12.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример

```
wlc# crypto generate pfx private-key tester.pem cert tester.crt ca default_ca.pem password
ascii-text 12345678 filename tester.p12
```

## crypto generate private-key

Команда предназначена для генерации приватного ключа пользователя, который необходим при формировании csr.

### Синтаксис

```
crypto generate private-key rsa [<RSA>] filename <NAME_KEY>
```

### Параметры

<RSA> – размер ключа в битах, принимает значения от 1024 до 4096;

<NAME\_KEY> – имя файла ключа, задается строкой до 31 символа, может содержать латинские буквы разного регистра, цифры и символы '.', '-', '\_', '@'.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример

```
wlc# crypto generate private-key rsa filename tester.pem
```

## update crypto default

Команда предназначена для регенерации дефолтных сертификатов системы.

### Синтаксис

```
update crypto default {ca | cert | dh} [valid-after <TIME> <DAY> <MONTH> <YEAR>]
[invalid-after <TIME> <DAY> <MONTH> <YEAR>]
```

### Параметры

ca – регенерация дефолтного корневого сертификата и его ключа;

cert – регенерация дефолтного сертификата устройства;

dh – регенерация ключа Диффи-Хеллмана;

valid-after – время и дата, после которых сертификат будет валиден;

invalid-after – время и дата, после которых сертификат будет невалиден;

<YEAR> – год, принимает значения [1970..2100];

<MONTH> – месяц, принимает значения [January/February/March/April/May/June/July/August/September/October/November/December];

<DAY> – день месяца, принимает значения [1..31];

<TIME> – время, задаётся в виде HH:MM:SS, где:

- HH – часы, принимает значение [0..23];



- MM – минуты, принимает значение [0..59];
- SS – секунды, принимает значение [0..59].

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример

```
wlc# update crypto default ca
```

```
wlc# update crypto default cert valid-after 00:03:00 01 June 2001 invalid-after 06:53:00 02
September 2006
```

## Управление профилями ap-models

- [ap-model](#)
- [object-group ap-models](#)

### ap-model

Команда предназначена для добавления моделей точек доступа в [object-group ap-models](#).

Использование отрицательной формы команды (no) удаляет профиль модель.

### Синтаксис

```
[no] ap-model <BOARD-TYPE>
```

### Параметры

<BOARD-TYPE> – модель точек доступа, допустимые значения:

- WEP-1L
- WEP-2L
- WEP-3L
- WEP-200L
- WEP-30L
- WEP-30L-Z
- WEP-3ax
- WOP-2L
- WOP-20L
- WOP-30L
- WOP-30LI
- WOP-30LS
- WEP-2ac
- WEP-2ac Smart
- WOP-2ac
- WOP-2ac:rev.B
- WOP-2ac:rev.C

При выполнении отрицательной формы команды со значением параметра "all" будут удалены все модели точек доступа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-OBJECT-GROUP-AP-MODELS

### Пример

```
wlc-30(config-object-group-ap-models)# ap-model WEP-1L
```

### object-group ap-models

Команда предназначена для создания профиля моделей точек доступа. Профили используются при настройке контроллера для ограничения работы сервисов по модели точки доступа.

Использование отрицательной формы команды (no) удаляет профиль.

### Синтаксис

```
[no] object-group ap-models <NAME>
```

### Параметры

<NAME> – имя конфигурируемого профиля моделей точек доступа, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра "all" будут удалены все профили моделей точек доступа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
wlc-30(config)# object-group ap-models Realtek
```

## 10 Настройка mDNS-reflector

- [clear ip mdns-reflector](#)
- [ip mdns-reflector](#)
- [ip mdns-reflector services](#)
- [show ip mdns-reflector](#)

### clear ip mdns-reflector

Данная команда обновляет данные в кэше mDNS-reflector.

#### Синтаксис

```
clear ip mdns-reflector
```

#### Параметры

Команда не содержит параметров.

#### Значение по умолчанию

Отсутствует.

#### Необходимый уровень привилегий

1

#### Командный режим

ROOT

#### Пример

```
esr# clear ip mdns-reflector
```

### ip mdns-reflector

Данная команда добавляет интерфейс в конфигурацию сервиса mDNS-reflector. Для запуска сервиса нужно включить в конфигурацию минимум два интерфейса.

Использование отрицательной команды (no) удаляет интерфейс из конфигурации сервиса mDNS-reflector.

#### Синтаксис

```
ip mdns-reflector  
no ip mdns-reflector
```

#### Параметры

Команда не содержит параметров.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

**Пример**

```
esr(config-if-gi)# ip mdns-reflector
```

**ip mdns-reflector services**

Данная команда активирует фильтрацию сервисов mDNS. Сервисы, указанные в object-group <NAME>, будут разрешены, все остальные будут запрещены. Если список фильтрации пуст, то будут доступны все сервисы.

Использование отрицательной команды (no) отключает фильтрацию сервисов mDNS.

**Синтаксис**

```
ip mdns-reflector services <OBJ-GROUP-URL-NAME>
no ip mdns-reflector services
```

**Параметры**

<OBJ-GROUP-URL-NAME> – имя профиля URL-адресов, задаётся строкой до 31 символа.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

## Командный режим

CONFIG

### Пример

```

esr (config)# object-group url test
esr (config-object-group-url)# url _ssh._tcp.local
esr (config-object-group-url)# exit
esr (config)#
esr (config)# ip mdns-reflector services test

```

## show ip mdns-reflector

Данная команда выводит список сервисов из кэша mDNS-reflector.

### Синтаксис

```
show ip mdns-reflector [ interfaces <IF> | service <NAME> ]
```

### Параметры

<IF> – интерфейс или группы интерфейсов, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<NAME> – имя профиля URL-адресов, задаётся строкой до 31 символа.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

1

## Командный режим

ROOT

### Пример

```

esr# show ip mdns-reflector
Interface IP address Hostname Service Port
-----
br21 172.21.0.10 WEP-12ac.local _brcm-sb._tcp 80
br21 172.21.0.10 WEP-12ac.local _telnet._tcp 23
br21 172.21.0.10 WEP-12ac.local _ssh._tcp 22
br21 172.21.0.10 WEP-12ac.local _http._tcp 80
br21 172.21.0.10 WEP-12ac.local _https._tcp 443
gi1/0/2.20 172.20.0.11 Book-tester-7.local _ssh._tcp 22
gi1/0/2.20 172.20.0.11 Book-tester-7.local _sftp-ssh._tcp 22
gi1/0/2.20 172.20.0.11 Book-tester-7.local _eppc._tcp 3031
gi1/0/2.20 172.20.0.11 Book-tester-7.local _rfb._tcp 5900
gi1/0/2.20 172.20.0.11 Book-tester-7.local _smb._tcp 445

```

```
gil/0/2.20 172.20.0.11 Book-tester-7.local _airplay._tcp 7000
gil/0/2.20 172.20.0.11 Book-tester-7.local _raop._tcp 5000
gil/0/2.20 172.20.0.11 Book-tester-7.local _teamviewer._tcp 2020
```

```
esr# show ip mdns-reflector service ssh
Interface IP address Hostname Service Port
```

```
-----
br21 172.21.0.10 WEP-12ac.local _ssh._tcp 22
gil/0/2.20 172.20.0.11 Book-tester-7.local _ssh._tcp 22
gil/0/2.20 172.20.0.11 Book-tester-7.local _sftp-ssh._tcp 22
```

## 11 Настройка общесистемных параметров


- Общие параметры
  - alias
  - hostname
  - set unit id
  - show cpu network-load
  - show cpu history
  - show cpu processes
  - show cpu utilization
  - show licence-manager status
  - show system
  - show system id
  - system fan-speed
  - system config-confirm timeout
  - system cpu load-balance overload-threshold
  - show unit id
  - system utilization softgre disable
  - update licence-manager licence
- Настройка подключения к серверу лицензирования
  - description
  - enable
  - host address
  - host port
  - ip vrf forwarding
  - licence-key
  - licence-manager
  - location
  - system-name

### Общие параметры

#### alias

Данная команда дает возможность настроить укороченные/специфичные команды в различных командных режимах.

Использование отрицательной формы команды (no) удаляет запись.

 Для активации настроенных alias необходимо переключиться к CLI маршрутизатора.

#### Синтаксис

```
alias {<ALIAS_NAME>} {<MODE>} {<COMMAND>}
no alias <ALIAS_NAME>
```

#### Параметры

<ALIAS\_NAME> – имя новой команды.

<MODE> – названия командного режима, в котором присутствует новая команда:

- aaa-ldap-profile-configure;
- config;
- config-aaa-das-profile;
- config-aaa-radius-profile;

- config-access-profile;
- config-acl;
- config-acl-rule;
- config-address-assignment-pool;
- config-antispam-profile;
- config-antispam-profile-rule;
- config-archive;
- config-bgp;
- config-bgp-af;
- config-bgp-aggregate;
- config-bgp-group;
- config-bgp-group-af;
- config-bgp-group-af-vrf;
- config-bgp-listen;
- config-bgp-local-as;
- config-bgp-neighbor;
- config-bgp-neighbor-af;
- config-bgp-neighbor-af-vrf;
- config-bgp-vrf;
- config-bgp-vrf-af;
- config-bgp-vrf-aggregate;
- config-bgp-vrf-group;
- config-bgp-vrf-listen;
- config-bgp-vrf-local-as;
- config-bgp-vrf-neighbor;
- config-bridge;
- config-cellular-modem;
- config-cellular-profile;
- config-class-map;
- config-class-policy-map;
- config-cluster;
- config-cluster-unit;
- config-content-provider;
- config-crypto-sync;
- config-das-server;
- config-dhcp-server;
- config-dhcp-server-failover;
- config-dhcp-server-vendor-specific;
- config-dhcp-vendor-id;
- config-dnat;
- config-dnat-pool;
- config-dnat-rule;
- config-dnat-ruleset;
- config-failover;
- config-firewall-failover;
- config-gre;
- config-http-profile;
- config-if-e1;
- config-if-gi;
- config-if-te;
- config-if-twe;
- config-if-fo;
- cpnfig-if-hu;
- config-if-loopback;
- config-if-multilink;
- config-if-port-channel;



- config-if-qinq;
- config-if-serial;
- config-if-sub;
- config-ike-gw;
- config-ike-keyring;
- config-ike-policy;
- config-ike-proposal;
- config-ip4ip4;
- config-ips;
- config-ips-auto-upgrade;
- config-ips-category;
- config-ips-category-rule;
- config-ips-category-rule-advanced;
- config-ips-content-filter;
- config-ips-policy;
- config-ips-policy-vendor;
- config-ips-policy-vendor-category;
- config-ips-upgrade-user-server;
- config-ipsec-policy;
- config-ipsec-proposal;
- config-ipsec-vpn;
- config-ipv6-dhcp-server;
- config-ipv6-ospf;
- config-ipv6-ospf-area;
- config-ipv6-ospf-vlink;
- config-ipv6-pl;
- config-ipv6-wan-rule;
- config-ipv6-wan-target;
- config-ipv6-wan-target-list;
- config-isis;
- config-key-chain;
- config-key-chain-key;
- config-l2tp;
- config-l2tp-server;
- config-l2tpv3;
- config-l2vpn;
- config-l2vpn-autodiscovery-bgp;
- config-l2vpn-eompls;
- config-l2vpn-pw;
- config-l2vpn-pw-class;
- config-ldap-server;
- config-ldp;
- config-ldp-af;
- config-ldp-af-if;
- config-ldp-neighbor;
- config-licence-manager;
- config-line-aux;
- config-line-console;
- config-line-ssh;
- config-line-telnet;
- config-lt;
- config-mailserver;
- config-mailserver-domain;
- config-mpls;
- config-net-policy;
- config-netflow-host;

- config-ntp;
- config-object-group-address-port;
- config-object-group-application;
- config-object-group-cf-kaspersky;
- config-object-group-content-filter;
- config-object-group-email;
- config-object-group-mac;
- config-object-group-network;
- config-object-group-service;
- config-object-group-url;
- config-openvpn;
- config-openvpn-server;
- config-openvpn-server-user;
- config-ospf;
- config-ospf-area;
- config-ospf-vlink;
- config-pl;
- config-policy-map;
- config-ppp-user;
- config-pppoe;
- config-pptp;
- config-pptp-server;
- config-profile;
- config-radius-server;
- config-rip;
- config-ripng;
- config-route-map;
- config-route-map-rule;
- config-security-zone;
- config-security-zone-pair;
- config-security-zone-pair-rule;
- config-sflow-host;
- config-sla-responder;
- config-sla-test;
- config-snat;
- config-snat-pool;
- config-snat-rule;
- config-snat-ruleset;
- config-snmp;
- config-snmp-host;
- config-snmp-user;
- config-subscriber-control;
- config-subscriber-default-service;
- config-subtunnel;
- config-syslog-destination;
- config-syslog-file;
- config-syslog-host;
- config-tacacs-server;
- config-tftp-server;
- config-track;
- config-user;
- config-vlan;
- config-vrf;
- config-vti;
- config-wan-rule;
- config-wan-target;

- config-wan-target-list;
- config-wireguard;
- config-wireguard-server;
- config-wireguard-server-peer;
- config-wireguard-tunnel-peer;
- config-wlc-failover;
- config-zabbix-agent;
- config-zabbix-proxy;
- debug;
- root;
- root-change-expired-password.

<COMMAND> – строка размером 96 символов. Команда CLI. Вводится полностью, без сокращений. Для ввода команды из нескольких слов обрамляется двойными кавычками.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример:

```
esr(config)# alias qwe root "show version"
```

### hostname

Команда позволяет назначить сетевое имя для маршрутизатора.

Использование отрицательной формы команды (no) устанавливает имя маршрутизатора по умолчанию.

### Синтаксис

```
hostname <NAME> [ unit <ID> ]
```

```
no hostname [ unit <ID> ]
```

### Параметры

<NAME> – сетевое имя маршрутизатора, задаётся строкой до 63 символов;

<ID> – номер юнита, принимает значения [1..2].

### Необходимый уровень привилегий

10

### Значение по умолчанию

esr-10/esr-12v/esr-12vf/esr-15/esr-15r/esr-15vf/esr-20/esr-21/esr-30/esr-31/esr-100/esr-200/esr-1000/esr-1200/esr-1500/esr-1511/esr-1700/esr-3100/esr-3200/esr-3200L/esr-3300/vesr – в зависимости от модели маршрутизатора.

## Командный режим

CONFIG

### Пример:

```
esr(config)# hostname router-1.eltex-co.ru
```

## set unit id

Данной командой выставляется номер юнита устройства.

### Синтаксис

```
set unit id <ID>
```

### Параметры

<ID> – номер юнита, принимает значения [1..2].

### Необходимый уровень привилегий

10

## Командный режим

ROOT

### Пример

```
esr# set unit id 2  
Unit ID will be 2 after reboot
```

## show cpu network-load

Данной командой осуществляется просмотр нагрузки, производимой сетевым трафиком.

### Синтаксис

```
show cpu network-load
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

1

**Командный режим**

ROOT

**Пример:**

```

esr# show cpu network-load
CPU ID      CPU load      Heaviest session      Session
-----      -
0           0             --                     0
1           49413        80.88.157.57 ->     9826
                172.129.22.57
2           46812        80.88.157.75 ->     9895
                172.129.22.75
3           49229        172.129.22.41 ->    9851
                80.88.157.41
4           0             --                     0
5           53019        80.88.157.77 ->     9989
                172.129.22.77
6           39699        80.88.157.79 ->     9863
                172.129.22.79
7           49726        172.129.22.45 ->    9804
                80.88.157.45
8           39789        172.129.22.61 ->    9779
                80.88.157.61
9           36876        80.88.157.59 ->     9775
                172.129.22.59
10          53041        172.129.22.5 ->     9679
                80.88.157.5
11          49010        172.129.22.47 ->    9896
                80.88.157.47
12          53082        172.129.22.13 ->   9650
                80.88.157.13
13          63027        80.88.157.69 ->    9617
                172.129.22.69
14          52722        80.88.157.73 ->   10001
                172.129.22.73
15          55165        80.88.157.71 ->    9924

```

- CPU ID – номер ядра CPU;
- CPU load – загрузка ядра CPU (pps);
- Heaviest session – сессия, которая нагружает ядро CPU больше всего;
- Session weight – утилизация сессии (pps).

**show cpu history**

Данной командой осуществляется просмотр истории использования ресурсов CPU.

**Синтаксис**

```
show cpu history [ average | max ] [cpu {<CPU>}] [timer {<TIMER>}]
```

## Параметры

average | max – указывает, историю какой статистики необходимо выводить, усредненную или максимальную за интервал. Без указания данного ключа выводится история усредненной статистики (average).

<CPU> – возможно указать номер конкретного CPU, по которому будет выводиться история статистики использования ресурсов CPU. Без указания номера CPU будет выводиться информация по всем CPU, используемым системой.

<TIMER> – в качестве параметров для данного ключа могут выступать:

- hours – отображает историю за последние 72 часа;
- minutes – отображает историю за последние 60 минут;
- seconds – отображает историю за последние 60 секунд;
- При отсутствии ключа timer выводятся 3 таблицы истории статистики использования CPU.

## Необходимый уровень привилегий

5

## Командный режим

ROOT

## Пример:

```
esr# show cpu history max cpu 0
CPU0
Last 60 seconds:
utilization, %
100
 90 #
 80 #
 70 #
 60 #
 50 ##
 40 ##
 30 # ##
 20 # ##
 10 #####
....|....|..
0 25 50
time, 5 sec.
Last 60 minutes:
utilization, %
100 #
 90 #
 80 #
 70 # #
 60 ## #
 50 ## #
 40 ## #
 30 ## # # # # #
 20 ## # # # # #
 10 #####
....|....|....|....|....|....|....|....|....|....|....|....|....|
0 5 10 15 20 25 30 35 40 45 50 55 60
time, min.
```

```

Last 72 hours:
utilization, %
100 #          # #          ##
 90 #          # ###         ##
 80 #          # ###         ##
 70 #          #####        ##
 60 #          #####        ##
 50 #          #####        ##
 40 #          #####        ##
 30 #####
 20 #####
 10 #####
  ....|....|...|...|...|...|...|...|...|...|...|...|...|...|..
  0  5 10 15 20 25 30 35 40 45 50 55 60 65 70
                        time, hours

```

## show cpu processes

Данной командой осуществляется просмотр использования ресурсов CPU-процессами.

### Синтаксис

```
show cpu processes [ active | name <PROCESS_NAME> ]
```

### Параметры

active – список активных процессов;

name – имя конкретного процесса.

### Необходимый уровень привилегий

15

### Командный режим

ROOT

### Пример:

```

esr# show cpu processes
PID      Name          CPU 5s   CPU 1m   CPU 5m   Memory   Runtime
-----  -
1384     Routing       0.00%   0.13%   0.13%   0.32%   00,00:45:46
 474     Modem-cfgmgr  0.00%   0.00%   0.00%   0.30%   00,00:00:56
 724     ESRFS         0.00%   0.02%   0.00%   0.10%   00,00:00:57
 447     File-mgr      0.00%   0.00%   0.00%   0.72%   00,00:02:51
 620     Modem-mgr     0.00%   0.00%   0.00%   0.32%   00,00:00:18
 472     IPC-mgr       0.00%   0.00%   0.00%   0.08%   00,00:00:45
 468     E1D           0.00%   0.00%   0.00%   0.12%   00,00:00:31
 305     SYSLOG        0.00%   0.00%   0.01%   0.23%   00,00:02:07
 297     HAVEGE        0.00%   0.00%   0.01%   0.14%   00,00:03:58
 239     Device-mgr    0.00%   0.02%   0.01%   0.11%   00,00:01:13
 319     RNGd          0.00%   0.02%   0.02%   0.03%   00,00:07:00
 371     Service-mgr  0.00%   0.02%   0.01%   0.14%   00,00:03:45
 363     Cp-mgr        0.00%   0.00%   0.00%   0.13%   00,00:01:25

```

366	Session-mgr	0.00%	0.00%	0.00%	0.14%	00,00:01:05
373	REXd	0.00%	0.00%	0.00%	0.17%	00,00:00:53
367	Lb	0.00%	0.00%	0.00%	0.07%	00,00:00:18
368	Alarm-mgr	0.00%	0.00%	0.00%	0.09%	00,00:00:43
359	Syslog-mgr	0.00%	0.00%	0.00%	0.16%	00,00:01:25
358	Ipc-hub	0.00%	0.13%	0.14%	0.23%	00,00:51:46
372	AAA-mgr	0.00%	0.00%	0.00%	0.23%	00,00:00:42
370	Systemdb	0.00%	0.02%	0.03%	0.15%	00,00:09:11
374	SFP-mgr	0.00%	0.02%	0.00%	0.22%	00,00:01:20
362	If-mgr-ng	0.00%	0.00%	0.00%	0.27%	00,00:01:33
369	Env-mgr	0.00%	0.03%	0.04%	0.23%	00,00:15:00
360	Lic-mgr	0.00%	0.03%	0.04%	0.24%	00,00:14:03
361	If-mgr	0.80%	0.60%	0.65%	0.36%	00,03:54:54
365	Cfgsync-mgr	0.00%	0.07%	0.02%	10.85%	00,00:37:08
364	Oi-mgr	0.20%	0.08%	0.08%	3.09%	00,00:31:08
31093	CLI	0.00%	0.10%	0.02%	0.58%	00,00:00:01

## show cpu utilization

Данной командой осуществляется просмотр использования ресурсов CPU.

### Синтаксис

```
show cpu utilization
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример:

```
esr# show cpu utilization
CPU   Last      Last      Last
      5 sec    1 min    5 min
----  -
0     1.98%    6.75%    20.02%
1     67.50%   15.62%    6.88%
2     65.43%   15.53%    6.94%
3     69.29%   16.08%    7.08%
4     89.90%   20.79%    9.14%
5     74.95%   17.14%    7.49%
6     87.61%   20.18%    8.85%
7     87.41%   20.17%    8.85%
8     81.84%   19.03%    8.40%
9     84.82%   19.79%    8.73%
10    84.53%   19.78%    8.75%
11    83.02%   19.40%    8.58%
12    83.73%   19.55%    8.63%
13    76.56%   16.99%    7.25%
```



14	70.47%	16.00%	6.95%
15	68.39%	15.07%	6.40%

## show licence-manager status

Данной командой осуществляется просмотр информации о взаимодействии с сервером ELM.

### Синтаксис

```
show licence-manager status
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

5

### Командный режим

ROOT

### Пример:

```
esr# show licence-manager status
ELM server type:          root
Last request status:     success
Last request to licence server: 2024-08-12 08:57:07
Next request to licence server: 2024-08-12 09:57:07
```

## show system

Данной командой осуществляется просмотр параметров окружения устройства.

### Синтаксис

```
show system
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

**Пример:**

```
wlc-30-failover# show system
System type:           Eltex WLC-30 Service Router
System name:           WLC-30
Software version:      1.30.0 build 11[f23466fadf] (2024-12-05 11:01:03)
Hardware version:      1v4
System uptime (d,h:m:s): 02,00:40:55
System MAC address:    CC:9D:A2:71:90:70
System serial number:  NP1F000053

  Temperature Table
  ~~~~~
-----
Temperature, C  CPU          Board          SFP          Coprocessor 1
-----
44              37              32              47

  Memory Table
  ~~~~~
-----
Total, MB      Used, MB      Free, MB
-----
RAM            3986.06      3048.75 (77%)  937.31 (23%)
FLASH          119.96       1.78 (2%)     118.18 (98%)
DATA           6068.10      462.11 (8%)   5605.99 (92%)
```

**show system id**

Данной командой осуществляется просмотр серийного номера устройства.

**Синтаксис**

```
show system id
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

1


**Командный режим**

ROOT

**Пример:**

```
esr# show system id
Serial number:
NP01000023
```

## system fan-speed

 Данная команда поддерживается только на маршрутизаторах ESR-100/200/1000/1200/1500/1511/1700.

Команда определяет режим работы системы охлаждения.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
system fan-speed { auto | max }  
no system fan-speed
```

### Параметры

auto – режим автоматического регулирования;

max – режим максимального охлаждения.

### Значение по умолчанию

auto – для ESR-100/200/1500/1511/1700;

max – для ESR-1000/1200.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример:

```
esr(config)# system fan-speed auto
```

## system config-confirm timeout

Данная команда определяет интервал времени ожидания для подтверждения текущей конфигурации. Если конфигурация не будет подтверждена, то после истечения времени ожидания произойдет откат на предыдущую примененную конфигурацию.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
system config-confirm timeout <TIME>  
no system config-confirm timeout
```

**Параметры**

<TIME> – интервал времени ожидания подтверждения резервирования конфигурации, принимает значение в секундах [120..86400].

**Значение по умолчанию**

600

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример:**

```
esr(config)# system config-confirm timeout 1200
```

**system cpu load-balance overload-threshold**

Данная команда определяет пороговое значение нагрузки ядра CPU (CPU load), при котором запускается механизм перебалансировки сессий между ядрами CPU.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
system cpu load-balance overload-threshold <CPU-LOAD>
```

```
no system cpu load-balance overload-threshold
```

**Параметры**

<CPU\_LOAD> – значение порога срабатывания перебалансировки [1000-200000].

**Значение по умолчанию**

30000 – для ESR-1x,

60000 – для остальных моделей ESR.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример:**

```
esr(config)# system cpu load-balance overload-threshold 15000
```

**show unit id**

Данной командой просматривается номер текущего юнита устройства, а также номер юнита устройства, который будет применен после перезагрузки.

**Синтаксис**

```
show unit id
```

**Необходимый уровень привилегий**

10

**Командный режим**

ROOT

**Пример**

```
esr# show unit id
Unit ID is 1
Unit ID will be 1 after reboot
```

**system utilization softgre disable**

Данная команда используется для отключения подсчета загруженности SoftGRE-туннелей. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
[no] system utilization softgre disable
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Подсчет включен.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример:**

```
esr(config)# system utilization softgre disable
```

**update licence-manager licence**

Данная команда позволяет принудительно отправить запрос на сервер ELM для получения актуальной информации о лицензии, не дожидаясь таймеров.

**Синтаксис**

```
update licence-manager licence
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

5

**Командный режим**

ROOT

**Пример:**

```
esr# update licence-manager licence
```

**Настройка подключения к серверу лицензирования****description**

Данной командой выполняется изменение описания.

Использование отрицательной формы команды (no) удаляет описание.

**Синтаксис**

```
[no] description <DESCRIPTION>
```

**Параметры**

<DESCRIPTION> – описание, задаётся строкой до 255 символов.

**Необходимый уровень привилегий**

10

## Командный режим

CONFIG-LICENCE-MANAGER

### Пример:

```
esr(config-licence-manager)# description protected-server
```

### enable

Данной командой включается менеджер лицензирования.

Использование отрицательной формы команды (no) отключает его.

### Синтаксис

[no] enable

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

15

## Командный режим

CONFIG-LICENCE-MANAGER

### Пример:

```
esr(config-licence-manager)# enable
```

### host address

Данной командой задаётся адрес сервера лицензирования.

### Синтаксис

host address { <ADDR> | <IPV6-ADDR> | <WORD> }

### Параметры

<ADDR> – IP-адрес сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес сервера, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<WORD> – DNS-имя сервера, задаётся строкой до 31 символа.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-LICENCE-MANAGER

## Пример

```
esr(config-licence-manager)# host address elm.eltex-co.ru
```

## host port

Данной командой задаётся номер TCP-порта сервера лицензирования.

Использование отрицательной формы команды (no) удаляет номер TCP-порта.

## Синтаксис

```
host port <PORT>  
no host port
```

## Параметры

<PORT> – номер TCP-порта, принимает значения [1..65535].

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-LICENCE-MANAGER

## Пример

```
esr(config-licence-manager)# host port 8098
```

## ip vrf forwarding

Данной командой задаётся имя экземпляра VRF, в котором будет работать менеджер лицензирования.

Использование отрицательной формы команды (no) удаляет привязку к экземпляру VRF.

## Синтаксис

```
[no] ip vrf forwarding <VRF>
```

## Параметры

<VRF> – имя VRF, задаётся строкой до 31 символов.



**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-LICENCE-MANAGER


**Пример**

```
vesr(config-licence-manager)# ip vrf forwarding LOCAL_VRF
```

**licence-key**

Данной командой задаётся лицензионный ключ vESR для аутентификации на сервере ELM.

Использование отрицательной формы команды (no) удаляет этот ключ.

 Данный параметр необходимо указывать только на устройстве vWLC.

**Синтаксис**

```
licence-key <KEY>  
no licence-key
```

**Параметры**

<KEY> – лицензионный ключ, задаётся строкой до 128 символов.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-LICENCE-MANAGER

**Пример**

```
vesr(config-licence-manager)# licence-key 1BC29B36F623BA82AAF6724FD3B16718
```

**licence-manager**

Данной командой осуществляется переход в режим конфигурирования менеджера лицензирования.

**Синтаксис**

```
licence-manager
```

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# licence-manager
```

## location

Данной командой задаётся текстовое описание, которое передаётся на сервер ELM.  
Использование отрицательной формы команды (no) удаляет это описание.

## Синтаксис

```
location <WORD>  
no location
```

## Параметры

<WORD> – описание, задаётся строкой до 255 символов.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-LICENCE-MANAGER

## Пример

```
esr(config-licence-manager)# location "Server room in Novokuznetsk office <3"
```

## system-name

Данной командой задаётся текстовое имя устройства, которое передаётся на сервер ELM.  
Использование отрицательной формы команды (no) присваивает этому имени значение по умолчанию.

## Синтаксис

```
system-name <WORD>  
no system-name
```

**Параметры**

<WORD> – имя, задаётся строкой до 253 символа.

**Значение по умолчанию**

По умолчанию значение system-name совпадает с hostname.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-LICENCE-MANAGER

**Пример**

```
esr(config-licence-manager)# system-name main-office
```

## 12 Управление системными часами

- `clock timezone`
- `burst`
- `iburst`
- `key`
- `maxpoll`
- `minpoll`
- `ntp access-addresses`
- `ntp authentication enable`
- `ntp authentication key-chain`
- `ntp authentication trusted-key`
- `ntp broadcast-client enable`
- `ntp dscp`
- `ntp enable`
- `ntp ipv6 source address`
- `ntp object-group query-only`
- `ntp object-group serve-only`
- `ntp peer`
- `ntp server`
- `ntp pool`
- `ntp source address`
- `prefer`
- `set date`
- `show date`
- `show ntp configuration`
- `show ntp peers`
- `version`

### clock timezone

Данной командой устанавливается часовой пояс.

Использование отрицательной формы команды (`no`) устанавливает часовой пояс по умолчанию.

### Синтаксис

```
clock timezone <OFFSET>
```

```
no clock timezone
```

### Параметры

<OFFSET> – обозначение зоны, содержащее сдвиг в часах относительно Greenwich Mean Time, принимает значения [gmt -12 .. gmt +12].

### Значение по умолчанию

gmt 0

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

**Пример:**

```
esr(config)# clock timezone gmt +7
```

**burst**

Данной командой включается отправка нескольких пакетов вместо одного при установлении соединения (на публичных серверах параметр не рекомендуется использовать).

Использование отрицательной формы команды (no) выключает отставку нескольких пакетов при установлении соединения.

**Синтаксис**

[no] burst

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Выключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-NTP

**Пример:**

```
esr(config-ntp)# burst
```

**iburst**

Данной командой включается отправка нескольких пакетов вместо одного в случае разрыва соединения.

Использование отрицательной формы команды (no) выключает отставку нескольких пакетов при разрыве соединения.

**Синтаксис**

[no] iburst

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Выключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-NTP

**Пример:**

```
esr(config-ntp)# iburst
```

**key**

Данной командой определяется ключ из списка доверенных ключей.

Использование отрицательной формы команды (no) удаляет привязку к указанному ключу.

**Синтаксис**

```
key <ID>
```

```
no key
```

**Параметры**

<ID> – идентификатор ключа, задается в диапазоне [1..255].

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-NTP

**Пример:**

```
esr(config-ntp)# key 245
```

**maxpoll**

Данная команда устанавливает максимальное значение интервала времени между отправкой сообщений NTP-серверу.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
maxpoll <INTERVAL>
```

```
no maxpoll
```

## Параметры

<INTERVAL> – максимальное значение интервала опроса. Параметр команды используется как показатель степени двойки при вычислении длительности интервала в секундах, вычисляется путем возведения двойки в степень, заданную параметром команды, принимает значение [4..17].

## Значение по умолчанию

10 ( $2^{10}$  = 1024 секунды или 17 минут 4 секунды).

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-NTP

## Пример:

```
esr(ntp-remote)# maxpoll 11
```

## minpoll

Данная команда устанавливает минимальное значение интервала времени между отправкой сообщений NTP-серверу.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
minpoll <INTERVAL>
```

```
no minpoll
```

## Параметры

<INTERVAL> – минимальное значение интервала опроса в секундах, вычисляется путем возведения двойки в степень, заданную параметром команды, принимает значение [1..6].

## Значение по умолчанию

6 ( $2^6$  = 64 секунды или 1 минута 4 секунды).

## Необходимый уровень привилегий

10

## Командный режим

### CONFIG-NTP

#### Пример:

```
esr(ntp-remote)# minpoll 4
```

## ntp access-addresses

Данной командой определяется список доверенных IP-адресов, с которыми может происходить обмен ntp-пакетами.

Использование отрицательной формы команды (no) удаляет указанный список.

#### Синтаксис

```
ntp access-addresses <NAME>
```

```
no ntp access-addresses
```

#### Параметры

<NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа.

#### Значение по умолчанию

Фильтрация отключена.

#### Необходимый уровень привилегий

15

## Командный режим

### CONFIG

#### Пример:

```
object-group network TIME
  ip prefix 10.0.0.0/25
  ipv6 prefix 2001::/64
exit

esr(config)# ntp access-addresses TIME
```

## ntp authentication enable

Данная команда включает аутентификацию для NTP-протокола.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.



**Синтаксис**

```
[no] ntp authentication enable
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Выключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример:**

```
esr(config)# ntp authentication enable
```

**ntp authentication key-chain**

Данная команда определяет набор паролей для аутентификации через алгоритм хеширования md5 с сервером, пиром.

Использование отрицательной формы команды (no) удаляет привязку к набору паролей.

**Синтаксис**

```
ntp authentication key-chain <KEYCHAIN>
```

```
no ntp authentication key-chain
```

**Параметры**

<KEYCHAIN> – идентификатор списка ключей, задаётся строкой до 16 символов.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример:**

```
esr(config)# ntp authentication key-chain lock
```

## **ntp authentication trusted-key**

Данной командой определяется список доверенных ключей из набора ключей. Использование отрицательной формы команды (no) удаляет указанный ключ.

### **Синтаксис**

```
[no] ntp authentication trusted-key <ID>
```

### **Параметры**

<ID> – идентификатор ключа, задается в диапазоне [1..255].

### **Необходимый уровень привилегий**

15

### **Командный режим**

CONFIG

### **Пример:**

```
esr(config)# ntp authentication trusted-key 25
```

## **ntp broadcast-client enable**

Данной командой включается режим приёма широковещательных сообщений NTP-серверов для глобальной конфигурации и всех существующих VRF. Маршрутизатор работает в качестве NTP-клиента. Если в конфигурации устройства заданы NTP-пиры и серверы, то в широковещательном режиме они игнорируются.

Использование отрицательной формы команды (no) выключает широковещательный режим.

### **Синтаксис**

```
[no] ntp broadcast-client enable
```

### **Параметры**

Команда не содержит параметров.

### **Значение по умолчанию**

Выключен.

### **Необходимый уровень привилегий**

10

### **Командный режим**

CONFIG

**Пример:**

```
esr(config)# ntp broadcast-client enable
```

**ntp dscp**

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов NTP-сервера.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

**Синтаксис**

```
ntp dscp <DSCP>
```

```
no ntp dscp
```

**Параметры**

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

**Значение по умолчанию**

46

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример:**

```
esr(config)# ntp dscp 40
```

**ntp enable**

Данной командой включается синхронизация системных часов с удаленными серверами по протоколу NTP для глобальной конфигурации и всех созданных VRF.

Использование отрицательной формы команды (no) выключает синхронизацию по протоколу NTP.

**Синтаксис**

```
[no] ntp enable
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Выключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример:**

```
esr(config)# ntp enable
```

**ntp ipv6 source address**

Данная команда используется для указания source-IPv6-адреса для NTP-пакетов для всех peer.

Использование отрицательной формы команды (no) устанавливает source-IPv6-адрес для NTP-пакетов по умолчанию.

**Синтаксис**

```
ntp ipv6 source address <IPV6-ADDR>
```

```
no ntp ipv6 source address
```

**Параметры**

<IPV6-ADDR> – IPv6-адрес RADIUS-сервера, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

**Значение по умолчанию**

IPv6-адрес интерфейса, с которого отправляется NTP-пакет.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример:**

```
esr(config)# ntp ipv6 source address fc00::1
```

## **ntp object-group query-only**

Данная команда включает режим query-only, ограничивающий взаимодействие по NTP для определенного профиля IP-адресов, входящих в эту группу. В этом режиме применяются следующие ограничения: запрещена синхронизация времени между одноранговыми пирами. Ответы на NTP-запросы (отдача времени) и обмен control messages разрешены.

Использование отрицательной формы команды (no) отключает данные ограничения.

### **Синтаксис**

```
ntp object-group query-only <NAME>  
no ntp object-group query-only
```

### **Параметры**

<NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа.

### **Значение по умолчанию**

Работа с NTP-пирами ничем не ограничивается.

### **Необходимый уровень привилегий**

15

### **Командный режим**

CONFIG

### **Пример:**

```
esr(config)# ntp object-group query-only OG-NTP-Q0
```

## **ntp object-group serve-only**

Данная команда включает режим serve-only, ограничивающий взаимодействие по NTP для определенного профиля IP-адресов, входящих в эту группу. В этом режиме применяются следующие ограничения: запрещен обмен control messages и синхронизация времени между одноранговыми пирами. Ответы на ntp-запросы (отдача времени) разрешены.

Использование отрицательной формы команды (no) отключает данные ограничения.

### **Синтаксис**

```
ntp object-group serve-only <NAME>  
no ntp object-group serve-only
```

### **Параметры**

<NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа.

**Значение по умолчанию**

Работа с NTP-пирами ничем не ограничивается.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример:**

```
esr(config)# ntp object-group serve-only OG-NTP-S0
```

**ntp peer**

Данная команда используется для установления партнерских отношений между NTP-серверами и перехода в командный режим CONFIG-NTP-PEER.

NTP-сервер на маршрутизаторе работает в режиме двусторонней активности с удаленным NTP-сервером, указанным в команде. В случае потери связи одного из партнеров с вышестоящим NTP-сервером он сможет синхронизировать время по серверу-партнеру.

Использование отрицательной формы команды (no) удаляет заданного NTP-партнера.

**Синтаксис**

```
[no] ntp peer { <ADDR> | <NAME> | <IPV6-ADDR> } [ vrf <VRF> ]
```

**Параметры**

<ADDR> – IP-адрес партнера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<NAME> – DNS-имя сервера, задаётся строкой до 31 символа;

<IPV6-ADDR> – IPv6-адрес партнера, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать NTP-партнер.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример:**

```
esr(config)# ntp peer 10.100.100.1
```

```
esr(config-ntp-peer)#
```

## ntp server

Данная команда используется для создания NTP-сервера и перехода в командный режим CONFIG-NTP-SERVER.

Маршрутизатор работает с указанным NTP-сервером в режиме односторонней активности. В данном режиме локальные часы маршрутизатора могут синхронизироваться с удаленным NTP-сервером.

Использование отрицательной формы команды (no) удаляет заданный NTP-сервер.

### Синтаксис

```
[no] ntp server { <ADDR> | <NAME> | <IPV6-ADDR> } [ vrf <VRF> ]
```

### Параметры

<ADDR> – IP-адрес сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<NAME> – DNS-имя сервера, задаётся строкой до 31 символа;

<IPV6-ADDR> – IPv6-адрес сервера, задаётся в виде X:X:X:X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать NTP-сервер.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример:

```
esr(config)# ntp server 10.100.100.2
esr(config-ntp-server)#
```

## ntp pool

Данная команда используется для создания NTP-pool и перехода в командный режим CONFIG-NTP-POOL.

Директива pool позволяет динамически получать IP-адреса серверов из указанного пула. Когда серверы становятся недоступными, они отбрасываются, а новые серверы добавляются на лету, поэтому в любой момент можно успешно опросить определенное количество серверов.

Использование отрицательной формы команды (no) удаляет заданный NTP-pool.

### Синтаксис

```
[no] ntp pool { <ADDR> | <NAME> | <IPV6-ADDR> } [ vrf <VRF> ]
```

## Параметры

<ADDR> – IP-адрес сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<NAME> – DNS-имя сервера, задаётся строкой до 31 символа;

<IPV6-ADDR> – IPv6-адрес сервера, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать NTP-сервер.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример:

```
esr(config)# ntp pool 10.100.100.3
esr(config-ntp-pool)#
```

## ntp source address

Данная команда используется для указания source-IP-адреса для NTP-пакетов для всех реер.

Использование отрицательной формы команды (no) устанавливает source-IP-адрес для NTP-пакетов по умолчанию.

## Синтаксис

```
ntp source address <ADDR>
```

```
no ntp source address
```

## Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

## Значение по умолчанию

IP-адрес интерфейса, с которого отправляется NTP-пакет.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG



**Пример:**

```
esr(config)# ntp source address 10.100.100.2
```

**prefer**

Команда отмечает данный NTP-сервер как предпочтительный при выборе из списка доступных (активных) NTP-серверов. Данная команда предоставляет возможность повысить приоритет при выборе NTP-сервера, но это никак не повлияет на работу текущей синхронизации.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
[no] prefer
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-NTP

**Пример:**

```
esr(ntp-remote)# prefer
```

**set date**

Данной командой устанавливается вручную системное время и дата.

**Синтаксис**

```
set date <TIME> [<DAY> <MONTH> [ <YEAR> ] ]
```

**Параметры**

<TIME> – устанавливаемое системное время, задаётся в виде HH:MM:SS, где:

- HH – часы, принимает значение [0..23];
- MM – минуты, принимает значение [0..59];
- SS – секунды, принимает значение [0..59].

<DAY> – день месяца, принимает значения [1..31];

<MONTH> – месяц, принимает значения [January/February/March/April/May/June/July/August/September/October/November/December];

<YEAR> – год, принимает значения [2001..2037].

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример:

```
esr# set date 16:35:00 15 May 2014
```

### show date

Данная команда позволяет посмотреть текущие системное время и дату.

### Синтаксис

```
show date
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример:

```
esr# show date  
2024-07-01 09:14:09"
```

### show ntp configuration

Данная команда отображает действующую (RUNNING) конфигурацию протокола NTP.

### Синтаксис

```
show ntp configuration
```

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

10

## Командный режим

ROOT

## Пример:

```

esr# show ntp configuration
NTP status:      Enabled
NTP mode:        client/server
NTP DSCP:        46

Address
Version  Min poll  Max poll  Prefer  burst  VRF instance  Type
-----  -
0.pool.ntp.org  6      10      No      Yes     No      --      pool      4
1.pool.ntp.org  6      10      No      No      Yes     --      server    4
2.pool.ntp.org  6      10      No      No      No      --      peer      4

```

## show ntp peers

Данная команда позволяет посмотреть текущее состояние NTP-серверов (пиров). В следующих таблицах приведен перечень отображаемых параметров и их описание.

Таблица 6 – Состояние удаленного сервера (пира)

Параметр	Описание
vrf	Имя экземпляра VRF.
remote	DNS-имя или IP-адрес сервера (пира). Первый символ в таблице используется для обозначения состояния сервера (пира), состояния описаны в <a href="#">таблице 7</a> .
refid	Идентификатор связи или IP-адрес того, с кем синхронизирован удаленный сервер (пир). Типы идентификаторов связи описаны в <a href="#">таблице 8</a> .
st	Стратум.
t	Отношение маршрутизатора к удаленному серверу (пиру), типы описаны в <a href="#">таблице 9</a> .
when	Период времени с момента, когда сервер (пир) последний раз опрашивался, в секундах ("h" часы, "d" дни).

Параметр	Описание
poll	Частота опроса сервера (пира).
reach	Восьмибитный сдвигаемый влево регистр, содержащий результаты опросов (1 = успешно, 0 = неуспешно), отображается в восьмеричной системе счисления.
delay	Время прохождения пакета до сервера (пира) и обратно, в миллисекундах.
offset	Среднее постоянное смещение времени маршрутизатора относительно сервера (пира).
jitter	Средний разброс отклонения времени (джиттер).

Таблица 7 – Состояние удаленного сервера (пира)

Тип	Описание
пробел	Указывает на то, что: <ul style="list-style-type: none"> <li>• не было ответов от удаленного сервера (пира);</li> <li>• сервер не используется, так как стратум имеет большое значение;</li> <li>• сервер (пир) использует данный маршрутизатор для синхронизации своих часов.</li> </ul>
x	Сервер (пир) не используется для синхронизации времени, отброшен алгоритмом пересечения.
-	Сервер (пир) не используется для синхронизации времени, отброшен кластерным алгоритмом.
#	Рабочий удаленный сервер (пир), но не используется, так как не вошел в число первых шести серверов (пиров), отсортированных по расстоянию синхронизации, является резервным.
+	Рабочий и предпочитаемый удаленный сервер (пир), включен алгоритмом объединения.
*	Сервер (пир), который в настоящее время является первичным источником времени.

Таблица 8 – Типы идентификаторов соединения с удаленным сервером (пиром)

Тип	Описание
.ACST.	Manycast-сервер.
.AUTH.	Ошибка аутентификации.
.AUTO.	Ошибка последовательности автоматического ключа.
.BCST.	Broadcast-сервер.
.CRYPT.	Ошибка протокола автоматического ключа.
.DENY.	Сервер отказал в доступе.

Тип	Описание
.INIT.	Инициализация соединения с сервером.
.MCST.	Multicast-сервер.
.TIME.	Таймаут соединения с сервером.
.STEP.	Ступенчатое изменение времени, смещение меньше предельного порога (1000 миллисекунд), но больше, чем шаг порога (125 миллисекунд).
.RATE.	Превышение частоты опросов.

Таблица 9 – Типы отношений маршрутизатора к удаленному серверу (пиру)

Тип	Описание
u	Unicast или manycast-клиент.
b	Broadcast или unicast-клиент.
p	Директива pool.
s	Двусторонняя связь (пир).
A	Manycast-сервер.
B	Broadcast-сервер.
M	Multicast-сервер.

**Синтаксис**

```
show ntp peers
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример:**

```
esr# sh ntp peers
Clock is synchronized, stratum 4, reference is 162.159.200.123
```

	remote					vrf	refid	st	t
when	poll	reach	delay	offset	jitter				
*	162.159.200.123						10.87.12.180	3	u 0
8	15	43.681	39.153	0.608					
+	91.207.136.55						89.109.251.22	2	u 2
8	15	61.094	42.103	0.182					
+	162.159.200.1						10.87.12.180	3	u 3
8	15	51.740	35.247	4.176					
	0.ru.pool.ntp.org						POOL	16	p 0
8	0	0.000	0.000	0.001					

## version

Данной командой устанавливается версия NTP-протокола.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
version <VERSION>
```

```
no version
```

## Параметры

<VERSION> – версия NTP-протокола, принимает значения [1..4].

## Значение по умолчанию

4

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-NTP

## Пример:

```
esr(ntp-remote)# version 3
```

## 13 Настройка AAA

- `aaa accounting commands`
- `aaa accounting login`
- `aaa authentication attempts max-fail`
- `aaa authentication enable`
- `aaa authentication login`
- `aaa authentication mode`
- `aaa authorization commands`
- `aaa authorization control-commands enable`
- `aaa authorization mode`
- `aaa das-profile`
- `aaa disable`
- `aaa radius-profile`
- `acct-port`
- `auth-port`
- `clear users blocked`
- `clear user-session`
- `clients`
- `commands authorization`
- `das-server`
- `das-server`
- `dead-interval`
- `description`
- `disable`
- `enable`
- `enable authentication`
- `enable password`
- `exec-timeout`
- `ip sftp enable`
- `key`
- `ldap-server base-dn`
- `ldap-server bind authenticate root-dn`
- `ldap-server bind authenticate root-password`
- `ldap-server bind timeout`
- `ldap-server dscp`
- `ldap-server host`
- `ldap-server naming-attribute`
- `ldap-server privilege-level-attribute`
- `ldap-server search filter user-object-class`
- `ldap-server search scope`
- `ldap-server search timeout`
- `ldap-server ssl crypto`
- `ldap-server ssl check-peer disable`
- `ldap-server ssl enable`
- `line`
- `login authentication`
- `password`
- `port`
- `priority`
- `privilege`
- `privilege`
- `radius-server dscp`
- `radius-server host`
- `radius-server host`
- `radius-server retransmit`
- `radius-server timeout`

- retransmit
- security passwords default-expired
- security passwords history
- security passwords lifetime
- security passwords lower-case
- security passwords max-length
- security passwords min-length
- security passwords numeric-count
- security passwords special-case
- security passwords symbol-types
- security passwords upper-case
- security snmp-community max-length
- security snmp-community min-length
- show aaa accounting
- show aaa authentication
- show aaa ldap-servers
- show aaa radius-servers
- show aaa tacacs-servers
- show users
- show users accounts
- show users blocked
- source-address
- source-interface
- system configuration-exclusively
- tacacs-server dscp
- tacacs-server host
- tacacs-server timeout
- tech-support login enable
- timeout
- usage
- username

## aaa accounting commands

Данной командой конфигурируется список способов учета команд, введенных в CLI.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
aaa accounting commands stop-only <METHOD>
```

```
no aaa accounting commands stop-only
```

### Параметры

<METHOD> – способы учета:

- tacacs – учет введенных команд по протоколу TACACS.

### Значение по умолчанию

Учёт не ведется.

### Необходимый уровень привилегий

15



## Командный режим

CONFIG

### Пример

```
esr(config)# aaa accounting commands stop-only tacacs
```

## aaa accounting login

Данной командой конфигурируется список способов учета сессий пользователей. Ведение учета активируется и прекращается, когда пользователь входит и отключается от системы, что соответствует значениям «start» и «stop» в сообщениях протоколов RADIUS и TACACS.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
aaa accounting login start-stop <METHOD 1> [ <METHOD 2> ]
```

```
no aaa accounting login start-stop
```

### Параметры

<METHOD> – способы учета:

- tacacs – учет сессий по протоколу TACACS;
- radius – учет сессий по протоколу RADIUS.

### Значение по умолчанию

Учет сессий ведется в локальный журнал.

### Необходимый уровень привилегий

15

## Командный режим

CONFIG

### Пример

```
esr(config)# aaa accounting login start-stop tacacs
```

## aaa authentication attempts max-fail

Данной командой устанавливается максимальное количество неудачных попыток аутентификации до блокировки пользователя и время, на которое происходит блокировка.

Использование отрицательной формы команды (no), значения количества попыток и период блокировки устанавливает по умолчанию.

## Синтаксис

```
aaa authentication attempts max-fail <COUNT> <TIME>
no aaa authentication attempts max-fail
```

## Параметры

<COUNT> – количество неудачных попыток аутентификации, после которых произойдет блокировка пользователя, принимает значения [1..65535];

<TIME> – интервал времени в секундах, на который будет заблокирован пользователь, принимает значения [1..65535].

## Значение по умолчанию

Количество неудачных попыток – 5.

Период блокировки – 300.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# aaa authentication attempts max-fail 5 30
```

## aaa authentication enable

Данной командой создаются списки способов аутентификации повышения привилегий пользователей. При неудачной попытке аутентификации по одному способу происходит попытка аутентификации по следующему способу в списке.

В конфигурации по умолчанию существует список с именем «default». Список «default» содержит один способ аутентификации – «enable». Чтобы использовать список для аутентификации повышения привилегий пользователей, необходимо выполнить его привязку командой, описанной в разделе [enable authentication](#).

Использование отрицательной формы команды (no) удаляет список способов аутентификации.

## Синтаксис

```
aaa authentication enable <NAME> <METHOD 1> [ <METHOD 2> ] [ <METHOD 3> ] [ <METHOD 4> ]
no aaa authentication enable <NAME>
```

## Параметры

<NAME> – имя списка: строка до 31 символа;

- default – имя списка «default».

<METHOD> – способы аутентификации:

- enable – аутентификация с помощью enable-паролей;
- tacacs – аутентификация по протоколу TACACS;
- radius – аутентификация по протоколу RADIUS;
- ldap – аутентификация по протоколу LDAP.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# aaa authentication enable enable-test tacacs enable
```

## aaa authentication login

Данной командой создаются списки способов аутентификации входа пользователей в систему. При неудачной попытке аутентификации по одному способу происходит попытка аутентификации по следующему в списке.

В конфигурации по умолчанию существует список с именем «default», данный список содержит один способ аутентификации – «local». Чтобы использовать список для аутентификации входа пользователей, необходимо выполнить его активацию командой, описанной в разделе [login authentication](#).

Использование отрицательной формы команды (no) удаляет список способов аутентификации.

## Синтаксис

```
aaa authentication login { default | <NAME> } <METHOD 1> [ <METHOD 2> ] [ <METHOD 3> ] [ <METHOD 4> ]
```

```
no aaa authentication login { default | <NAME> }
```

## Параметры

<NAME> – имя списка, задаётся строкой до 31 символа.

## Способы аутентификации:

- local – аутентификация с помощью локальной базы пользователей;
- tacacs – аутентификация по списку TACACS-серверов;
- radius – аутентификация по списку RADIUS-серверов;
- ldap – аутентификация по списку LDAP-серверов.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# aaa authentication login login-test tacacs local
```

## aaa authentication mode

Данной командой определяется режим работы со списками методов аутентификации.

Использование отрицательной формы команды (no) удаляет список способов аутентификации.

## Синтаксис

```
[no] aaa authentication mode { break | chain }
```

## Параметры

**break** – при аутентификации будут использоваться последующие методы в случае недоступности более приоритетного;

**chain** – при аутентификации будут использоваться последующие методы в случае получения отказа от более приоритетного.

## Значение по умолчанию

chain

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# aaa authentication mode break
```

## aaa authorization commands

Данной командой создаются списки способов авторизации команд, вводимых пользователем в систему. При неудачной попытке авторизации по одному способу происходит попытка аутентификации по следующему способу в списке.

В конфигурации по умолчанию существует список с именем «default», данный список содержит один способ авторизации – «local». Чтобы использовать список для авторизации команд вводимых пользователем в систему, необходимо выполнить его активацию командой, описанной в разделе `commands authorization`.

Использование отрицательной формы команды (no) удаляет список способов авторизации.

**Синтаксис**

```
aaa authorization commands { default | <NAME> } <METHOD 1> [ <METHOD 2> ]
no aaa authorization commands { default | <NAME> }
```

**Параметры**

<NAME> – имя списка, задаётся строкой до 31 символа.

**Способы аутентификации:**

- local – авторизация с помощью локальной базы пользователей;
- tacacs – авторизация по списку TACACS-серверов.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# aaa authorization commands
```

**aaa authorization control-commands enable**

Данной командой включается авторизация контрольных команд (help, logout, end, exit) на TACACS-сервере. В конфигурации по умолчанию эта функция отключена.

Использование отрицательной формы команды (no) возвращает состояние к дефолтному.

**Синтаксис**

```
[ no ] aaa authorization control-commands enable
```

**Значение по умолчанию**

no aaa authorization control-commands enable

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

## Пример

```
esr(config)# aaa authorization control-commands enable
```

## aaa authorization mode

Данной командой определяется режим работы со списками методов авторизации.

Использование отрицательной формы команды (no) удаляет список способов авторизации.

## Синтаксис

```
[no] aaa authorization mode { break | chain }
```

## Параметры

**break** – при аутентификации будут использоваться последующие методы в случае недоступности более приоритетного;

**chain** – при аутентификации будут использоваться последующие методы в случае получения отказа от более приоритетного.

## Значение по умолчанию

chain

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# aaa authorization mode break
```

## aaa das-profile

Данная команда используется для добавления профиля серверов динамической авторизации (DAS) и перехода в командный режим DAS SERVER PROFILE.

Использование отрицательной формы команды (no) удаляет заданный профиль серверов динамической авторизации (DAS).

## Синтаксис

```
[no] aaa das-profile <NAME>
```

## Параметры

<NAME> – имя профиля серверов динамической авторизации (DAS), задается строкой до 31 символа.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# aaa das-profile profile1
esr(config-aaa-das-profile)#
```

## aaa disable

Данной командой отключает доступ на маршрутизатор через консольный интерфейс.

При использовании отрицательной формы команды (no) доступ на маршрутизатор через консольный интерфейс включается.

## Синтаксис

[no] aaa disable

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Доступ на маршрутизатор через консольный интерфейс включен.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-LINE-CONSOLE

## Пример

```
esr(config-line-console)# aaa disable
```

## aaa radius-profile

Данная команда используется для добавления профиля RADIUS-серверов и перехода в командный режим RADIUS SERVER PROFILE.

Использование отрицательной формы команды (no) удаляет заданный профиль RADIUS-серверов.

## Синтаксис

```
[no] aaa radius-profile <NAME>
```

## Параметры

<NAME> – имя профиля RADIUS-серверов, задается строкой до 31 символа.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# aaa radius-profile profile1  
esr(config-aaa-radius-profile)#
```

## acct-port

Данной командой задается номер порта для обмена данными с удаленным RADIUS-сервером при выполнении аккаунтинга.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
acct-port <PORT>
```

```
no acct-port
```

## Параметры

<PORT> – номер UDP-порта для обмена данными с удаленным сервером, принимает значения [1..65535].

## Значение по умолчанию

1813

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-RADIUS-SERVER

## Пример

```
esr(config-radius-server)# acct-port 4444
```



## auth-port

Данной командой задаётся номер порта для обмена данными с удаленным RADIUS-сервером при выполнении аутентификации и авторизации.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
auth-port <PORT>
```

```
no auth-port
```

### Параметры

<PORT> – номер UDP-порта для обмена данными с удаленным сервером, принимает значения [1..65535].

### Значение по умолчанию

1812

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-RADIUS-SERVER

### Пример

```
esr(config-radius-server)# auth-port 4444
```

## clear users blocked

Данной командой удаляется информация о неправильных попытках аутентификации различных пользователей.

### Синтаксис

```
clear users blocked <NAME>
```

### Параметры

<NAME> – имя пользователя, для которого необходимо очистить статистику неправильных попыток аутентификации, задаётся строкой до 31 символа.

Без указания имени пользователя очищается вся таблица неправильных попыток аутентификации.

### Необходимый уровень привилегий

15

## Командный режим

ROOT

### Пример

```
esr# clear users blocked
```

## clear user-session

Данной командой закрывается рабочая сессия пользователя CLI.

### Синтаксис

```
clear user-session [ <USERNAME> | <SESSION> ]
```

### Параметры

<NAME> – имя пользователя сессию которого необходимо закрыть, задаётся строкой до 31 символа.

<SESSION> – номер терминальной сессии которую необходимо закрыть, задаётся число в диапазоне [1..10].

### Необходимый уровень привилегий

15

## Командный режим

ROOT

### Пример

```
esr# clear users-session
```

## clients

Данной командой определяется список клиентов динамической авторизации (DAC), на запросы которых будет отвечать сервер динамической авторизации (DAS).

Использование отрицательной формы команды (no) удаляет список клиентов динамической авторизации (DAC).

### Синтаксис

```
clients object-group <NAME>
```

```
no clients
```

### Параметры

<NAME> – имя профиля IP-адресов, содержащий адреса клиентов динамической авторизации, задаётся строкой до 31 символа.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-DAS-SERVER

**Пример**

```
esr(config-das-server)# clients object-group pcrf
```

**commands authorization**

Данной командой осуществляется активация списка авторизации команд вводимых пользователем в систему.

В конфигурации по умолчанию активным является список с именем «default», данный список содержит один способ авторизации – «local».

Использование отрицательной формы команды (no) делает список с именем «default» активным.

**Синтаксис**

```
commands authorization <NAME>
```

```
no commands authorization
```

**Параметры**

<NAME> – имя списка, задаётся строкой до 31 символа.

**Значение по умолчанию**

default

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-LINE-CONSOLE

CONFIG-LINE-TELNET

CONFIG-LINE-SSH

**Пример**

```
esr(config-line-ssh)# commands authorization authorization-test
```

**das-server**

Данная команда используется для добавления сервера динамической авторизации (DAS) и перехода в командный режим DAS SERVER. Сервера динамической авторизации (DAS) принимают RADIUS CoA запросы от клиентов динамической авторизации (DAC), например, отключение или повторный запрос списка сервисов пользователя.

Использование отрицательной формы команды (no) удаляет заданный сервер динамической авторизации (DAS).

**Синтаксис**

```
[no] das-server <NAME>
```

**Параметры**

<NAME> – имя сервера динамической авторизации (DAS), задается строкой до 31 символа.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# das-server main
esr(config-das-server)#
```

**das-server**

Данная команда используется для добавления сервера динамической авторизации (DAS) в конфигурируемый профиль серверов динамической авторизации.

Использование отрицательной формы команды (no) удаляет заданный сервер динамической авторизации (DAS).

**Синтаксис**

```
[no] das-server <NAME>
```

**Параметры**

<NAME> – имя сервера динамической авторизации (DAS), задается строкой до 31 символа.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-AAA-DAS-PROFILE

## Пример

```
esr(config)# das-server main
esr(config-das-server)#
```

## dead-interval

Данной командой задаётся интервал, в течении которого на RADIUS-сервер не будут отправляться пакеты. В данное состояние RADIUS-сервер переводится по истечении таймаута ожидания ответа на запрос последнего допустимого повтора (см. раздел [radius-server retransmit](#)).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
dead-interval <SEC>
no dead-interval
```

## Параметры

<SEC> – период времени в секундах, принимает значения [0..3600].

## Значение по умолчанию

120

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-RADIUS-SERVER

## Пример

```
esr(config-radius-server)# dead-interval 600
```

## description

Команда используется для изменения описания профиля серверов динамической авторизации (DAS) или профиля RADIUS-серверов.

Использование отрицательной формы команды (no) удаляет описание профиля.

## Синтаксис

```
description <DESCRIPTION>
no description
```

## Параметры

<DESCRIPTION> – описание профиля, задаётся строкой до 255 символов.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-DAS-SERVER-PROFILE

CONFIG-RADIUS-SERVER-PROFILE

## Пример

Установить описание для профиля IP-адресов:

```
esr(config-aaa-das-profile)# description "Main profile"
```

## disable

Данной командой производится понижение уровня привилегий пользователя до первоначальных.

## Синтаксис

disable

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

2

## Командный режим

ROOT

## Пример

```
esr# disable
esr>
```

## enable

Данной командой производится повышение уровня привилегий пользователя. Способы аутентификации повышения привилегий пользователей задаются с помощью команды, описанной в разделе [aaa authentication attempts max-fail](#).

✘ По умолчанию в конфигурации установлен метод аутентификации по паролю «enable». При этом пароли не заданы, то есть любой системный пользователь может получить 15 необходимый уровень привилегий.

✘ Для аутентификации повышения привилегий по протоколам TACACS/RADIUS/LDAP на сервере должны быть созданы пользователи \$enab<PRIV>\$, где <PRIV> – необходимый уровень привилегий пользователя, который должен быть аутентифицирован.

### Синтаксис

enable [ <PRIV> ]

### Параметры

<PRIV> – необходимый уровень привилегий, принимает значение [2..15].

### Значение по умолчанию

15

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
esr> enable 10
esr#
```

## enable authentication

Данной командой осуществляется активация списка аутентификации повышения привилегий пользователей, который будет использоваться в конфигурируемом терминале.

В конфигурации по умолчанию активным является список с именем «default», данный список содержит один способ аутентификации – «enable».

Использование отрицательной формы команды (no) делает список с именем «default» активным.

### Синтаксис

```
enable authentication <NAME>
no enable authentication
```

### Параметры

<NAME> – имя списка, задаётся строкой до 31 символа.

**Значение по умолчанию**

default

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-LINE-CONSOLE

CONFIG-LINE-TELNET


CONFIG-LINE-SSH

**Пример**

```
esr(config-line-console)# enable authentication enable-test
```

**enable password**

Данной командой устанавливается пароль, который будет запрашиваться при повышении уровня привилегий пользователя.

 По умолчанию в конфигурации пароли не заданы, то есть любой системный пользователь может получить 15 необходимый уровень привилегий.

Использование отрицательной формы команды (no) удаляет пароль из системы.

**Синтаксис**

```
enable password { <CLEAR-TEXT> | encrypted <HASH_SHA512> } [ privilege <PRIV> ]
```

```
no enable password [ privilege <PRIV> ]
```

**Параметры**

<CLEAR-TEXT> – пароль, задаётся строкой [1..32] символов, принимает значения [0-9a-fA-F];

<HASH\_SHA512> – хеш пароля по алгоритму sha512, задаётся строкой из 110 символов;

<PRIV> – необходимый уровень привилегий, принимает значение [2..15], значение по умолчанию 15.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG



## Пример

```
esr(config)# enable password 12345678 privilege 10
```

## exec-timeout

Данной командой задаётся интервал, по истечении которого будет разрываться бездействующая сессия.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
exec-timeout <SEC>
```

```
no exec-timeout
```

## Параметры

<SEC> – период времени в минутах, принимает значения [1..65535].

## Значение по умолчанию

30 минут

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-LINE-CONSOLE


CONFIG-LINE-SSH

CONFIG-LINE-TELNET

CONFIG-LINE-AUX<sup>1</sup>

## Пример


```
esr(config-line-ssh)# exec-timeout 600
```

 <sup>1</sup> Только для ESR-21.

## ip sftp enable

Данной командой на маршрутизаторе включается доступ по sftp для конфигурируемого пользователя.

При использовании отрицательной формы команды (no) отключает доступ по sftp для конфигурируемого пользователя.

 При активации данной команды у пользователя пропадает доступ к CLI устройства.

**Синтаксис**

```
[no] ip sftp enable
```

**Параметры**

Отсутствуют

**Значение по умолчанию**

Выключено

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-USER

**Пример**

```
esr(config-user)# ip sftp enable
```

**key**

Данной командой задаётся пароль для аутентификации на удаленном сервере.

Использование отрицательной формы команды (no) удаляет заданный пароль для аутентификации на удаленном сервере.

**Синтаксис**

```
key ascii-text { <TEXT> | encrypted <ENCRYPTED-TEXT> }
```

```
no key
```

**Параметры**

<TEXT> – строка [8..16] ASCII-символов (для TACACS-сервера – [1..16] ASCII-символов);

<ENCRYPTED-TEXT> – зашифрованный пароль, размером [8..16] байт, задаётся строкой [16..32] символов (для TACACS-сервера – до 120 символов).

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-TACACS-SERVER

CONFIG-RADIUS-SERVER

CONFIG-DAS-SERVER

## Пример

```
esr(config-tacacs-server)# key ascii-text 12345678
```

## ldap-server base-dn

Данной командой задаётся базовый DN (Distinguished name), который будет использоваться при поиске пользователей.

Использование отрицательной формы команды (no) удаляет заданный базовый DN.

### Синтаксис

```
ldap-server base-dn <NAME>
```

```
no ldap-server base-dn
```

### Параметры

<NAME> – базовый DN, задается строкой до 255 символов.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

## Пример

```
esr(config)# ldap-server base-dn "dc=example,dc=com"
```

## ldap-server bind authenticate root-dn

Данной командой задаётся DN (Distinguished name) пользователя с правами администратора, под которым будет происходить авторизация на LDAP-сервере при поиске пользователей.

Использование отрицательной формы команды (no) удаляет заданный DN пользователя.

### Синтаксис

```
ldap-server bind authenticate root-dn <NAME>
```

```
no bind authenticate root-dn
```

### Параметры

<NAME> – DN пользователя с правами администратора, задается строкой до 255 символов.

### Необходимый уровень привилегий

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ldap-server bind authenticate root-dn "cn=admin,dc=example,dc=com"
```

**ldap-server bind authenticate root-password**

Данной командой задаётся пароль пользователя с правами администратора, под которым будет происходить авторизация на LDAP-сервере при поиске пользователей.

Использование отрицательной формы команды (no) удаляет заданный пароль пользователя.

**Синтаксис**

```
ldap-server bind authenticate root-password ascii-text { <TEXT> | encrypted <ENCRYPTED-TEXT> }
```

```
no bind authenticate root-password
```

**Параметры**

<TEXT> – строка [8..16] ASCII-символов;

<ENCRYPTED-TEXT> – зашифрованный пароль, размером [8..16] байт, задаётся строкой [16..32] символов.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ldap-server bind authenticate root-password ascii-text 12345678
```

**ldap-server bind timeout**

Данной командой задаётся интервал, по истечении которого устройство считает, что LDAP-сервер недоступен.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
ldap-server bind timeout <SEC>
```

```
no ldap-server bind timeout
```

**Параметры**

<SEC> – период времени в секундах, принимает значения [1..30].

**Значение по умолчанию**

3 секунды

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ldap-server bind timeout 5
```

**ldap-server dscp**

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов LDAP-сервера.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

**Синтаксис**

```
ldap-server dscp <DSCP>
```

```
no ldap-server dscp
```

**Параметры**

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

**Значение по умолчанию**

63

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ldap-server dscp 40
```

## ldap-server host

Данная команда используется для добавления LDAP-сервера в список используемых серверов и перехода в командный режим LDAP SERVER.

Использование отрицательной формы команды (no) удаляет заданный LDAP-сервер.

### Синтаксис

```
[no] ldap-server host { <ADDR> | <IPV6-ADDR> | <LDAP-HOST-NAME> } [ vrf <VRF> ]
```

### Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

<ADDR> – IP-адрес LDAP-сервера, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес LDAP-сервера, задается в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

<LDAP-HOST-NAME> – DNS-имя LDAP-сервера, задается строкой до 31 символа. Для корректного сопоставления DNS-имени хоста с IP-адресом на маршрутизаторе должна быть запущена и настроена функция [domain lookup enable](#).

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

### Пример

```
esr(config)# ldap-server host 10.100.100.1
esr(config-ldap-server)#
```

## ldap-server naming-attribute

Данной командой задается имя атрибута объекта, со значением которого идет сравнение имени искомого пользователя на LDAP-сервере.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ldap-server naming-attribute <NAME>
```

```
no ldap-server naming-attribute
```

### Параметры

<NAME> – имя атрибута объекта, задается строкой до 127 символов.

**Значение по умолчанию**

uid

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ldap-server naming-attribute displayName
```

**ldap-server privilege-level-attribute**

Данной командой задаётся имя атрибута объекта, значение которого будет определять начальные привилегии пользователя на устройстве. Атрибут должен принимать значения [1..15]. Если указанный атрибут отсутствует или содержит недопустимое значение, то начальные привилегии пользователя будут соответствовать привилегиям пользователя «remote».

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
ldap-server privilege-level-attribute <NAME>
```

```
no ldap-server privilege-level-attribute
```

**Параметры**

<NAME> – имя атрибута объекта, задаётся строкой до 127 символов.

**Значение по умолчанию**

priv-lvl

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ldap-server privilege-level-attribute title
```

## ldap-server search filter user-object-class

Данной командой задаётся имя класса объектов, среди которых необходимо выполнять поиск пользователей на LDAP-сервере.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ldap-server search filter user-object-class <NAME>
no ldap-server search filter user-object-class
```

### Параметры

<NAME> – имя класса объектов, задаётся строкой до 127 символов.

### Значение по умолчанию

posixAccount

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

### Пример

```
esr(config)# ldap-server search filter user-object-class shadowAccount
```

## ldap-server search scope

Данной командой задаётся область поиска пользователей в дереве LDAP-сервера.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ldap-server search scope <SCOPE>
no ldap-server search scope
```

### Параметры

<SCOPE> – область поиска пользователей на LDAP-сервере, принимает следующие значения:

- onelevel – выполнять поиск в объектах на следующем уровне после базового DN в дереве LDAP-сервера;
- subtree – выполнять поиск во всех объектах поддерева базового DN в дереве LDAP-сервера.

### Значение по умолчанию

subtree



## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# ldap-server search scope onelevel
```

## ldap-server search timeout

Данной командой задаётся интервал, по истечении которого устройство считает, что LDAP-сервер не нашел записей пользователей, подходящих под условие поиска.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
ldap-server search timeout <SEC>
```

```
no ldap-server search timeout
```

## Параметры

<SEC> – период времени в секундах, принимает значения [0..30].

## Значение по умолчанию

0 – устройство ожидает завершения поиска и получения ответа от LDAP-сервера.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# ldap-server search timeout 10
```

## ldap-server ssl crypto

Данной командой указывается сертификат, который будет использоваться для установления SSL-соединения с LDAP-сервером.

Использование отрицательной формы команды (no) удаляет из конфигурации сертификат для установления SSL-соединения с LDAP-сервером.

**Синтаксис**

```
ldap-server ssl crypto <FILE-NAME>  
no ldap-server ssl crypto
```

**Параметры**

Отсутствуют.

**Значение по умолчанию**

Сертификат не указан.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ldap-server ssl crypto ldap-ssl.crt
```

**ldap-server ssl check-peer disable**

Данной командой отключается верификация LDAP-сервера по сертификату.

Использование отрицательной формы команды (no) включает верификацию LDAP-сервера по сертификату.

**Синтаксис**

```
[no] ldap-server ssl check-peer disable
```

**Параметры**

Отсутствуют.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

## Пример

```
esr(config)# ldap-server ssl check-peer disable
```

## ldap-server ssl enable

Данной командой включается использование SSL-соединения для LDAP-подключения (LDAP over SSL). Использование отрицательной формы команды (no) отключается использование SSL-соединения для LDAP-подключения (LDAP over SSL).

## Синтаксис

```
[no] ldap-server ssl enable
```

## Параметры

Отсутствуют.

## Значение по умолчанию

Отключено.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# ldap-server ssl enable
```

## line

Данной командой осуществляется переход в режим конфигурирования соответствующего терминала: локальная консоль, удаленная консоль (Telnet), удаленная защищенная консоль (SSH).

Использование отрицательной формы команды (no) устанавливает параметры терминала по умолчанию. Параметры по умолчанию описаны в разделах [login authentication](#) и [enable authentication](#).

## Синтаксис

```
[no] line <TYPE>
```

## Параметры

<TYPE> – тип консоли:

- console – локальная консоль;
- telnet – удаленная консоль;
- ssh – защищенная удаленная консоль.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# line console
esr(config-line-console)#
```

## login authentication

Данной командой осуществляется активация списка аутентификации входа пользователей в систему, который будет использоваться в конфигурируемом терминале.

В конфигурации по умолчанию активным является список с именем «default», данный список содержит один способ аутентификации – «local».

Использование отрицательной формы команды (no) делает список с именем «default» активным.

## Синтаксис

```
login authentication <NAME>
```

```
no login authentication
```

## Параметры

<NAME> – имя списка, задается строкой до 31 символа.

## Значение по умолчанию

default

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-LINE-CONSOLE

CONFIG-LINE-TELNET

CONFIG-LINE-SSH

## Пример

```
esr(config-line-console)# login authentication login-test
```

## password

Команда для установки пароля определенному пользователю для входа в систему. Пароль может быть задан как в открытом виде, так и в виде хеш sha512.

Использование отрицательной формы команды (no) удаляет пароль пользователя из системы.

## Синтаксис

```
password { <CLEAR-TEXT> | encrypted <HASH_SHA512> }
```

```
no password
```

## Параметры

<CLEAR-TEXT> – пароль, задаётся строкой [1..32] символов, принимает значения [0-9a-fA-F];

<HASH\_SHA512> – хеш пароля по алгоритму sha512, задаётся строкой из 110 символов.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-USER

CHANGE-EXPIRED-PASSWORD

## Пример

```
esr(config-user) password test
```

## port

Данной командой задаётся номер порта для обмена данными с удаленным сервером.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
port <PORT>
```

```
no port
```

## Параметры

<PORT> – номер TCP/UDP-порта для обмена данными с удаленным сервером, принимает значения [1..65535].

**Значение по умолчанию**

49 – для TACACS-сервера;

389 – для LDAP-сервера;

Не установлено – для DAS-сервера.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-TACACS-SERVER

CONFIG-LDAP-SERVER

CONFIG-DAS-SERVER

**Пример**

```
esr(config-tacacs-server)# port 4444
```

**priority**

Данной командой задаётся приоритет использования удаленного сервера. Чем ниже значение, тем приоритетнее сервер.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

priority <PRIORITY>

no priority

**Параметры**

<PRIORITY> – приоритет использования удаленного сервера, принимает значения [1..65535].

**Значение по умолчанию**

1

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-TACACS-SERVER

CONFIG-RADIUS-SERVER

CONFIG-LDAP-SERVER

## Пример

```
esr(config-tacacs-server)# priority 5
```

## privilege

Данной командой производится установка уровня привилегий пользователя. Набор команд, который доступен пользователю, зависит от уровня привилегий. Пользователям с уровнями привилегий от 1 до 9 доступен только просмотр информации. Пользователям с уровнем привилегий от 10 до 15 доступна большая часть команд конфигурирования. Пользователям с уровнем привилегий 15 доступен полный набор команд. Требуемый необходимый уровень привилегий команд может быть изменен, описание в разделе [description](#).

Использование отрицательной формы команды (`no`) устанавливает необходимый уровень привилегий по умолчанию.

Назначение начального уровня привилегий пользователям происходит следующим образом:

- необходимый уровень привилегий пользователям из локальной базы назначается указанной командой;
- необходимый уровень привилегий для пользователей, авторизовавшихся по протоколу RADIUS, извлекается из атрибута `cisco-avpair = "shell:priv-lvl=<PRIV>"`;
- необходимый уровень привилегий для пользователей, авторизовавшихся по протоколу TACACS, извлекается из атрибута `priv-lvl=<PRIV>`;
- уровень привилегий для пользователей, авторизовавшихся по протоколу LDAP, извлекается из атрибута, заданного командой `privilege-level-attribute`, описанной в разделе [line](#), по умолчанию **priv-lvl=<PRIV>**.

Если при аутентификации пользователя через протоколы TACACS/RADIUS/LDAP не была получена вышеуказанная опция или была получена опция с некорректным значением, то пользователю будут назначены привилегии пользователя «remote», по умолчанию 1. Необходимый уровень привилегий пользователя «remote» можно изменить аналогично любому другому пользователю из локальной базы с помощью указанной команды.

## Синтаксис

```
privilege <PRIV>
no privilege
```

## Параметры

<PRIV> – необходимый уровень привилегий, принимает значение [1..15].

## Значение по умолчанию

1

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-USER

## Пример

```
esr(config-user)# privilege 15
```

## privilege

Данной командой производится установка минимального уровня привилегий пользователя, необходимого для выполнения команды из указанного поддерева команд.

Использование отрицательной формы команды (no) устанавливает необходимый уровень привилегий по умолчанию.

## Синтаксис

```
privilege <COMMAND-MODE> level <PRIV> <COMMAND>
```

```
no privilege <COMMAND-MODE> <COMMAND>
```

## Параметры

<COMMAND-MODE> – командный режим, может принимать значения:

- change-expired-password;
- config;
- config-aaa-das-profile;
- config-aaa-radius-profile;
- config-access-profile;
- config-acl;
- config-acl-rule;
- config-archive;
- config-bgp;
- config-bgp-af;
- config-bgp-group;
- config-bgp-neighbor;
- config-bridge;
- config-cellular-modem;
- config-cellular-profile;
- config-class-map;
- config-class-policy-map;
- config-das-server;
- config-dhcp-server;
- config-dhcp-vendor-id;
- config-dnat;
- config-dnat-pool;
- config-dnat-rule;
- config-dnat-ruleset;
- config-if-e1;
- config-gre;
- config-if-gi;
- config-if-te;
- config-if-twe;
- config-if-fo;
- config-if-hu;
- config-ike-gw;
- config-ike-policy;
- config-ike-proposal;



- config-ip4ip4;
- config-ipsec-policy;
- config-ipsec-proposal;
- config-ipsec-vpn;
- config-ipv6-bgp-af;
- config-ipv6-bgp-group;
- config-ipv6-bgp-neighbor;
- config-ipv6-dhcp-server;
- config-ipv6-ospf;
- config-ipv6-ospf-area;
- config-ipv6-ospf-vlink;
- config-ipv6-pl;
- config-ipv6-wan-rule;
- config-ipv6-wan-target;
- config-ipv6-wan-target-list;
- config-keychain;
- config-keychain-key;
- config-l2tp-server;
- config-l2tpv3;
- config-ldap-server;
- config-line-console;
- config-line-ssh;
- config-line-telnet;
- config-loopback;
- config-lt;
- config-mst;
- config-multilink;
- config-netflow-host;
- config-network-policy;
- config-ntp;
- config-object-group-application;
- config-object-group-mac;
- config-object-group-network;
- config-object-group-service;
- config-object-group-url;
- config-openvpn;
- config-openvpn-server;
- config-ospf;
- config-ospf-area;
- config-ospf-vlink;
- config-pl;
- config-policy-map;
- config-pool;
- config-port-channel;
- config-ppp-user;
- config-pppoe;
- config-pptp;
- config-pptp-server;
- config-profile;
- config-qinq-if;
- config-radius-server;
- config-rip;
- config-route-map;
- config-route-map-rule;
- config-service-port;
- config-sflow-host;
- config-snat;

- config-snat-pool;
- config-snat-rule;
- config-snat-ruleset;
- config-snmp-host;
- config-snmp-user;
- config-softgre;
- config-subif;
- config-subscriber-control;
- config-subscriber-default-service;
- config-subtunnel;
- config-tacacs-server;
- config-tracking;
- config-user;
- config-vlan;
- config-vrf;
- config-vti;
- config-wan-rule;
- config-wan-target;
- config-wan-target-list;
- config-wireless;
- config-zone;
- config-zone-pair;
- config-zone-pair-rule;
- debug;
- root.

<PRIV> – необходимый уровень привилегий, принимает значение [1..15];

<COMMAND> – поддерево команд, задается строкой до 255 символов.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

### Пример

Установить для поддерева команд «show» корневого командного режима необходимый уровень привилегий 2. Команды поддерева «show interfaces» оставить с уровнем привилегий 1.

```
esr(config)# privilege root level 2 "show"
esr(config)# privilege root level 1 "show interfaces"
```

### radius-server dscp

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов RADIUS-сервера.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

### Синтаксис

radius-server dscp <DSCP>

```
no radius-server dscp
```

### Параметры

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

### Значение по умолчанию

63

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# radius-server dscp 40
```

### radius-server host

Данная команда используется для добавления RADIUS-сервера в список используемых серверов и перехода в командный режим RADIUS SERVER.

Использование отрицательной формы команды (no) удаляет заданный RADIUS-сервер.

### Синтаксис

```
[no] radius-server host { <ADDR> | <IPV6-ADDR> } [ vrf <VRF> ]
```

### Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

<ADDR> – IP-адрес RADIUS-сервера, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес RADIUS-сервера, задается в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

### Пример

```
esr(config)# radius-server host 10.100.100.1
```

```
esr(config-radius-server)#
```

## radius-server host

Данная команда используется для добавления RADIUS-сервера в профиль RADIUS-серверов.

Использование отрицательной формы команды (no) удаляет заданный RADIUS-сервер из профиля.

### Синтаксис

```
[no] radius-server host { <ADDR> | <IPV6-ADDR> } [ vrf <VRF> ]
```

### Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

<ADDR> – IP-адрес RADIUS-сервера, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес RADIUS-сервера, задается в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-RADIUS-SERVER-PROFILE

### Пример

```
esr(config-aaa-radius-profile)# radius-server host 10.100.100.1
```

## radius-server retransmit

Данной командой задается количество перезапросов к последнему активному RADIUS-серверу, которое будет выполнено перед выполнением запросов к следующим RADIUS-серверам в списке.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
radius-server retransmit <COUNT>
```

```
no radius-server retransmit
```

### Параметры

<COUNT> – количество перезапросов к RADIUS-серверу, принимает значения [1..10].

### Значение по умолчанию

1

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# radius-server retransmit 5
```

**radius-server timeout**

Данной командой задаётся интервал, по истечении которого устройство считает, что RADIUS-сервер недоступен.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
radius-server timeout <SEC>
```

```
no radius-server timeout
```

**Параметры**

<SEC> – период времени в секундах, принимает значения [1..30].

**Значение по умолчанию**

3 секунды

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# radius-server timeout 5
```

**retransmit**

Данной командой задаётся количество перезапросов к RADIUS-серверу, которое будет выполнено перед выполнением запросов к следующим RADIUS-серверам в списке.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
retransmit <COUNT>  
no retransmit
```

**Параметры**

<COUNT> – количество перезапросов к RADIUS-серверу, принимает значения [1..10].

**Значение по умолчанию**

Не задан, используется значение глобального параметра, описанного в разделе [radius-server retransmit](#).

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-RADIUS-SERVER

**Пример**

```
esr(config)# retransmit 5
```

**security passwords default-expired**

Данной командой включается запрос на смену пароля по умолчанию для пользователя admin.

Использование отрицательной формы команды (no) отключает запрос на смену пароля по умолчанию.

**Синтаксис**

```
[no] security passwords default-expired
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Запрос на смену пароля по умолчанию отключен.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

## Пример

```
esr(config)# security passwords default-expired
```

## security passwords history

Данной командой включается режим запрета на использование ранее установленных паролей локальных пользователей. В качестве параметра указывается количество паролей, сохраняемых в памяти маршрутизатора.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
security passwords history <COUNT>
```

```
no security passwords history
```

## Параметры

<COUNT> – количество паролей, сохраняемых в памяти маршрутизатора [0..15]. При уменьшении данного значения лишние, более старые пароли удаляются.

## Значение по умолчанию

1

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# security passwords history 5
```

## security passwords lifetime

Данной командой устанавливается время действия пароля локального пользователя. При попытке подключения пользователя с истекшим паролем пользователь будет направлен в режим принудительной смены пароля.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
security passwords lifetime <TIME>
```

```
no security passwords lifetime
```

## Параметры

<TIME> – интервал времени действия пароля в днях, принимает значения [1..365].

## Значение по умолчанию

Время действия пароля локального пользователя не ограничено.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# security passwords lifetime 30
```

## security passwords lower-case

Данной командой устанавливается минимальное количество строчных букв в пароле локального пользователя, ENABLE-пароле, имени SMNPv3-пользователя и SMNPv1/SMNPv2 с комьюнити.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
security passwords lower-case <COUNT>
```

```
no security passwords lower-case
```

## Параметры

<COUNT> – минимальное количество строчных букв в пароле локального пользователя, ENABLE-пароле, имени SMNPv3-пользователя и SMNPv1/SMNPv2 с комьюнити [0..32].

## Значение по умолчанию

0

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# security passwords lower-case 2
```



## security passwords max-length

Данной командой устанавливается ограничение на максимальную длину пароля локального пользователя и ENABLE-пароля.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
security passwords max-length <NUM>
```

```
no security passwords max-length
```

### Параметры

<NUM> – максимальное количество символов в пароле, задается в диапазоне [1..32].

### Значение по умолчанию

32

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

### Пример

```
esr(config)# security passwords max-length 30
```

## security passwords min-length

Данной командой устанавливается ограничение на минимальную длину пароля локального пользователя и ENABLE-пароля.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
security passwords min-length <NUM>
```

```
no security passwords min-length
```

### Параметры

<NUM> – минимальное количество символов в пароле, задается в диапазоне [1..32].

### Значение по умолчанию

1

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# security passwords min-length 10
```

**security passwords numeric-count**

Данной командой устанавливается минимальное количество цифр в пароле локального пользователя, ENABLE-пароле, имени SNMPv3-пользователя и SNMPv1/SNMPv2 с комьюнити.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
security passwords numeric-count <COUNT>
```

```
no security passwords numeric-count
```

**Параметры**

<COUNT> – минимальное количество цифр в пароле локального пользователя, ENABLE-пароле, имени SNMPv3-пользователя и SNMPv1/SNMPv2 с комьюнити [0..32].

**Значение по умолчанию**

0

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# security passwords numeric-count 2
```

**security passwords special-case**

Данной командой устанавливается минимальное количество специальных символов в пароле локального пользователя, ENABLE-пароле, имени SNMPv3-пользователя и SNMPv1/SNMPv2 с комьюнити.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
security passwords special-case <COUNT>
no security passwords special-case
```

**Параметры**

<COUNT> – минимальное количество специальных символов в пароле локального пользователя, ENABLE-пароле, имени SNMPv3-пользователя и SNMPv1/SNMPv2 с комьюнити [0..32].

**Значение по умолчанию**

0

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# security passwords special-case 2
```

**security passwords symbol-types**

Данной командой устанавливается минимальное количество типов символов, которые должны присутствовать в пароле локального пользователя, ENABLE-пароле, имени SNMPv3-пользователя и SNMPv1/SNMPv2 с комьюнити.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
security passwords symbol-types <COUNT>
no security passwords symbol-types
```

**Параметры**

<COUNT> – минимальное количество типов символов в пароле локального пользователя, ENABLE-пароле, имени SNMPv3-пользователя и SNMPv1/SNMPv2 с комьюнити [1..4].

**Значение по умолчанию**

1

**Необходимый уровень привилегий**

15

## Командный режим

CONFIG

### Пример

```
esr(config)# security passwords symbol-types 2
```

### **security passwords upper-case**

Данной командой устанавливается минимальное количество прописных (заглавных) букв в пароле локального пользователя, ENABLE-пароле, имени SNMPv3-пользователя и SNMPv1/SNMPv2 с комьюнити.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
security passwords upper-case <COUNT>
```

```
no security passwords upper-case
```

### Параметры

<COUNT> – минимальное количество прописных (заглавных) букв в пароле локального пользователя, ENABLE-пароле, имени SNMPv3-пользователя и SNMPv1/SNMPv2 с комьюнити [0..32].

### Значение по умолчанию

0

### Необходимый уровень привилегий

15

## Командный режим

CONFIG

### Пример

```
esr(config)# security passwords upper-case 2
```

### **security snmp-community max-length**

Данной командой устанавливается ограничение на максимальную длину SNMPv1/SNMPv2 с комьюнити.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
security snmp-community max-length <NUM>
```

```
no security snmp-community max-length
```

### Параметры

<NUM> – максимальное количество символов в комьюнити, задается в диапазоне [1..128].

### Значение по умолчанию

128

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

### Пример

```
esr(config)# security snmp-community max-length 30
```

### security snmp-community min-length

Данной командой устанавливается ограничение на минимальную длину SMNPv1/SMNPv2 с комьюнити. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
security passwords min-length <NUM>
```

```
no security passwords min-length
```

### Параметры

<NUM> – минимальное количество символов в комьюнити, задается в диапазоне [1..128].

### Значение по умолчанию

1

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

### Пример

```
esr(config)# security snmp-community min-length 10
```

## show aaa accounting

Данная команда позволяет просмотреть настроенные параметры учета.

### Синтаксис

```
show aaa accounting
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример

```
esr# show aaa accounting
Login :          radius
Commands :      tacacs
```

## show aaa authentication

Данная команда позволяет просмотреть списки способов аутентификации пользователей, а также активные списки каждого типа терминалов.

### Синтаксис

```
show aaa authentication
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример

```
esr# show aaa authentication
Login Authentication Method Lists
~~~~~
```

```

List                Methods
-----
default           local
  Enable Authentication Method Lists
  ~~~~~
List                Methods
-----
default           enable
  Lines configuration
  ~~~~~
Line                Login method list                Enable method list
-----
console            default                default
telnet             default                default
ssh                default                default

```

## show aaa ldap-servers

Данная команда позволяет просмотреть параметры LDAP-серверов.

### Синтаксис

```
show aaa ldap-servers
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

15

### Командный режим

ROOT

### Пример

```

esr# show aaa ldap-servers
Base DN:                dc=example,dc=com
Root DN:                 cn=admin,dc=example,dc=com
Root password:          CDE65039E5591FA3
Naming attribute:       uid
Privilege level attribute: priv-lvl
User object class:      posixAccount
DSCP:                   63
Bind timeout:           3
Search timeout:         0
Search scope:           subtree
IP Address                Port                Priority
-----
10.100.100.1             389                 1

```

## show aaa radius-servers

Данная команда позволяет просмотреть параметры RADIUS-серверов.

**Синтаксис**

```
show aaa radius-servers
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

15

**Командный режим**

ROOT

**Пример**

```
esr# show aaa radius-servers
Timeout:      3
Retransmit:   1
DSCP:         63
IP Address    Timeout    Priority    Usage      Key
-----
2.2.2.2      --         1         all        9DA7076CA30B5FFE0DC9C4
2.4.4.4      --         1         all        9DA7076BA30B4EFCE5
```

**show aaa tacacs-servers**

Данная команда позволяет просмотреть параметры TACACS-серверов.

**Синтаксис**

```
show aaa tacacs-servers
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

15

**Командный режим**

ROOT

**Пример**

```
esr# show aaa tacacs-servers
Timeout :      3
DSCP:         63
IP Address    Port          Priority      Key
```



```

-----
10.100.100.1          49          1          CDE65039E5591FA3
10.100.100.5          49          10         CDE65039E5591FA3

```

**show users**

Данная команда позволяет просмотреть активные сессии пользователей системы.

**Синтаксис**

```
show users
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример**

```

esr# show users
SID      User name      Logged in at      Protocol      Host
Timers  Login/Priv    Level
-----  -
0 *      admin          2023-06-26 15:47:38  SSH           192.168.1.44
00:29:59/00:00:00  15
1        admin          2023-06-26 15:47:38  Telnet        192.168.1.44
00:25:08/00:00:00  15
2        admin          2023-06-26 15:47:38  Console       Console
00:29:56/00:00:00  15

```

**show users accounts**

Данная команда позволяет просмотреть конфигурацию пользователей системы.

**Синтаксис**

```
show users accounts
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

10

**Командный режим**

ROOT

**Пример**

```

esr# show user accounts
Name                               Password                               Privilege
-----                               -
admin                               $6$1sxrvGaV8Za8oX/K$YNe15xYPZ4cj    15
                                     bemYWYNpQBQKDXWE9v0aoKgQ
                                     kRCEb0EMNuus09Kmg7UBs7nA3buEM87e
                                     Eu.rA6tZq0
techsupport                          $6$YfwntIwU$ah7UxPZTemKhjpSWvVsV    15
                                     9jHcp. 9lweQaSldw7ZtUr
                                     uH66uZx9.EBASff//hUj80bUaC484TNR
                                     x.
remote                               $6$YfwntIwU$ah7UxPZTemKhjpSWvVsV    1
                                     9jHcp.kqFAK.vmvyY9lweQaSldw7ZtUr
                                     uH66uZx9.EBASff//hUj80bUaC484TNR
                                     x.
operator                             $6$eILpbbyRxedCzvVD$4RHP08mjXvNf    1
                                     urX7V/UlCZ1oHIWMwE6h5f
                                     zgwZQUZcPoZCEyaqQQqCicRMRuPwhxrQ
                                     bvGChWreW1

```

**show users blocked**

Данная команда позволяет просматривать список пользователей, для которых был введен неправильный пароль. Пользователь удаляется из списка после ввода правильного пароля при аутентификации.

**Синтаксис**

```
show users blocked [ <NAME> ]
```

**Параметры**

<NAME> – имя пользователя, для которого необходимо отобразить статистику неправильных попыток аутентификации, задаётся строкой до 31 символа.

Без указания имени пользователя отображается вся таблица неправильных попыток аутентификации.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример**

```

esr# show users blocked

```

User name	Failures	Latest failure	From
----- tester	----- 4	----- 2017-09-10 08:29:42	----- 0.0.0.0

## source-address

Данной командой определяется IPv4/IPv6-адрес маршрутизатора, который будет использоваться в качестве IPv4/IPv6-адреса источника в отправляемых пакетах на конфигурируемый AAA-сервер.

Использование отрицательной формы команды (no) удаляет указанный IPv4/IPv6-адрес источника.

### Синтаксис

```
source-address { <ADDR> | <IPV6-ADDR> | object-group <NETWORK_OBJ_GROUP_NAME> }
no source-address
```

### Параметры

<ADDR> – IP-адрес источника, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес источника, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

<NETWORK\_OBJ\_GROUP\_NAME> – список адресов, которые будут использоваться в качестве source address.

### Необходимый уровень привилегий

15

### Командный режим

```
CONFIG-RADIUS-SERVER
CONFIG-TACACS-SERVER
CONFIG-LDAP-SERVER
```

### Пример

```
esr(config-radius-server)# source-address 220::71
```

## source-interface

Данной командой определяется интерфейс или туннель маршрутизатора, IPv4/IPv6-адрес которого будет использоваться в качестве IPv4/IPv6-адреса источника в отправляемых пакетах на конфигурируемый AAA-сервер.

Использование отрицательной формы команды (no) удаляет указанный интерфейс или туннель.

### Синтаксис

```
source-interface { <IF> | <TUN> }
no source-interface
```

## Параметры

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#).

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-RADIUS-SERVER

CONFIG-TACACS-SERVER

CONFIG-LDAP-SERVER

## Пример

```
esr(config-radius-server)# source-interface gigabitethernet 1/0/1
```

## system configuration-exclusively

Данной командой включается ограничение количества сессий CLI до одной.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

## Синтаксис

```
[no] system configuration-exclusively
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Отключено.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# system configuration-exclusively
```

## **tacacs-server dscp**

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов TACACS-сервера.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

### **Синтаксис**

```
tacacs-server dscp <DSCP>
no tacacs-server dscp
```

### **Параметры**

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

### **Значение по умолчанию**

63

### **Необходимый уровень привилегий**

10

### **Командный режим**

CONFIG

### **Пример**

```
esr(config)# tacacs-server dscp 40
```

## **tacacs-server host**

Данная команда используется для добавления TACACS-сервера в список используемых серверов и перехода в командный режим TACACS SERVER.

Использование отрицательной формы команды (no) удаляет заданный TACACS-сервер.

### **Синтаксис**

```
[no] tacacs-server host { <ADDR> | <IPV6-ADDR> } [ vrf <VRF> ]
```

### **Параметры**

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

<ADDR> – IP-адрес TACACS-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

<IPV6-ADDR> – IPv6-адрес TACACS-сервера, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# tacacs-server host 10.100.100.1
esr(config-tacacs-server)#
```

**tacacs-server timeout**

Данной командой задаётся интервал, по истечении которого устройство считает, что TACACS-сервер недоступен.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
tacacs-server timeout <SEC>
```

```
no tacacs-server timeout
```

**Параметры**

<SEC> – период времени в секундах, принимает значения [1..30].

**Значение по умолчанию**

3 секунды.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# tacacs-server timeout 5
```

**tech-support login enable**

Данной командой включается низкоуровневый удаленный доступ к системе с помощью пользователя «techsupport». Низкоуровневый доступ к системе позволит получить технической поддержке всю необходимую информацию, когда это необходимо.

Использование отрицательной формы команды (no) выключает низкоуровневый удаленный доступ к системе с помощью пользователя «techsupport».

### Синтаксис

```
[no] tech-support login enable
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

### Пример

```
esr(config)# tech-support login enable
```

## timeout

Данной командой задаётся интервал, по истечении которого устройство считает, что RADIUS-сервер недоступен.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
timeout <SEC>  
no timeout
```

### Параметры

<SEC> – период времени в секундах, принимает значения [1..30].

### Значение по умолчанию

Не задан, используется значение глобального таймера, описанного в разделе [radius-server timeout](#).

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-RADIUS-SERVER

## Пример

```
esr(config-radius-server)# timeout 7
```

## usage

Данная команда определяет тип соединений для аутентификации которых будет использоваться RADIUS-сервера.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
usage { all | aaa | auth | acct | pptp | l2tp }
```

```
no usage
```

## Параметры

all – все типы соединений;

aaa – RADIUS-сервер будет использоваться для аутентификации, авторизации и учета telnet-, ssh-, console-сессий;

auth – RADIUS-сервер будет использоваться для аутентификации и авторизации telnet-, ssh-, console-сессий;

acct – RADIUS-сервер будет использоваться для учета telnet-, ssh-, console-сессий;

pptp – RADIUS-сервер будет использоваться для аутентификации, авторизации и учета удаленных пользователей, подключающихся по протоколу PPTP;

l2tp – RADIUS-сервер будет использоваться для аутентификации, авторизации и учета удаленных пользователей, подключающихся по протоколу L2TP over IPsec.

## Значение по умолчанию

all

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-RADIUS-SERVER

## Пример

```
esr(config-radius-server)# usage pptp
```

## username

Данной командой выполняется добавление пользователя в локальную базу пользователей и осуществляется переход в режим настройки параметров пользователя.



Использование отрицательной формы команды (no) удаляет пользователя из системы.

### Синтаксис

```
[no] username <NAME>
```

### Параметры

<NAME> – имя пользователя, задаётся строкой до 31 символа. Если использовать команду для удаления, то при указании значения «all» будут удалены все пользователи.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

### Пример

```
esr(config)# username test  
esr(config-user)#
```

## 14 Конфигурирование и мониторинг интерфейсов

- Общие команды
  - clear interfaces counters
  - description
  - history statistics
  - interface
  - ip tcp adjust-mss
  - load-average
  - mode
  - mtu
  - rate-limit arp-broadcast
  - rate-limit arp-broadcast pps
  - show interfaces counters
  - show interfaces description
  - show interfaces history
  - show interfaces status
  - show interfaces utilization
  - show system jumbo-frames
  - shutdown
  - speed
  - switchport dot1q ethertype egress stag
  - switchport mode
  - system jumbo-frames
- Физические интерфейсы
  - snmp trap link-status
  - show interfaces protected-ports
  - show interfaces sfp
  - show interfaces switch-port configuration
  - show interfaces switch-port status
  - threshold rx-utilization
  - threshold rx-utilization monitoring
  - threshold tx-utilization
  - threshold tx-utilization monitoring
  - switchport community
  - switchport protected
  - switchport protected-port
- Агрегированные интерфейсы
  - channel-group
  - lacp port-priority
  - lacp system-priority
  - lacp timeout
  - port-channel load-balance
  - show interfaces port-channel
  - show lacp counters
  - show lacp interfaces
  - show lacp parameters
- E1/multilink
  - enable
  - ip tcp header-compression
  - ip tcp compression-connections
  - mrru
  - password
  - ppp authentication chap
  - ppp chap hostname
  - ppp chap password
  - ppp chap refuse

- ppp chap username
- ppp ipcp accept-address
- ppp ipcp remote-address
- ppp max-configure
- ppp max-failure
- ppp max-terminate
- ppp mru
- ppp multilink
- ppp multilink-group
- ppp timeout keepalive
- ppp timeout retry
- show controllers e1
- switchport e1 slot
- switchport e1 clock source
- switchport e1 crc
- switchport e1 framing
- switchport e1 invert data
- switchport e1 linecode
- switchport e1 timeslots
- switchport e1 unframed
- switchport mode e1
- Последовательные интерфейсы
  - chat-script
  - clear line aux
  - databits
  - dialer
  - dialer idle-timeout
  - dialer in-band
  - dialer map
  - dialer string
  - flowcontrol
  - line aux
  - modem inout
  - parity
  - show line aux
  - speed
  - stopbits
  - transport telnet port
- Беспроводные модемы
  - allowed-auth
  - allowed-mode
  - apn
  - cellular modem
  - cellular profile
  - device
  - enable
  - ip-version
  - mode
  - mru
  - number
  - password
  - pin
  - preferred-mode
  - profile
  - show cellular configuration modem
  - show cellular configuration profile

- show cellular status modem
- user
- Настройка телефонных портов
  - authentication name
  - authentication password
  - call-forwarding busy
  - call-forwarding busy number
  - call-forwarding no-answer
  - call-forwarding no-answer number
  - call-forwarding no-answer timeout
  - call-forwarding unconditional
  - call-forwarding unconditional number
  - call-waiting
  - caller-id mode
  - dial-type
  - enable
  - flash call-transfer
  - flash call-transfer mode
  - flash timer
  - hotline
  - hotline ipt
  - hotline pstn
  - hotline number
  - hotline number ipt
  - hotline number pstn
  - hotline timeout
  - hotline timeout ipt
  - hotline timeout pstn
  - hybrid rx
  - hybrid tx
  - ipt prefix-name
  - ipt prefix-number
  - ipt offhook-ringing
  - ipt ring-number
  - profile sip
  - profile pbx
  - pstn transmit-number
  - pstn transmit-prefix
  - shutdown
  - sip port
  - sip user display-name
  - sip user phone
  - timing delay
  - timing digit
  - timing flash
  - timing pulse-digit
  - timing pulse-interdigit
  - timing pulse-pause
  - timing tone-digit
  - timing tone-interdigit
- Тестирование телефонных портов
  - test voice-port start
  - test voice-port status

Порядок именования интерфейсов маршрутизатора описан в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

Команды, введенные в режиме конфигурирования интерфейса (группы интерфейсов), применяются к выбранному интерфейсу (группе интерфейсов).

## Общие команды

### clear interfaces counters

Данной командой осуществляется сброс счетчиков заданного системного интерфейса или группы интерфейсов.

#### Синтаксис

```
clear interfaces counters [<IF>]
```

#### Параметры

<IF> – наименование системного интерфейса или группы интерфейсов, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

Можно указать несколько интерфейсов перечислением через «,» либо указать диапазон интерфейсов через «-». Если не указывать индексы интерфейсов, то будут очищены счетчики всех интерфейсов заданной группы. При выполнении команды без параметра будут очищены счетчики всех системных интерфейсов.

#### Необходимый уровень привилегий

10

#### Командный режим

ROOT

#### Пример

```
esr# clear interfaces counters gigabitethernet 1/0/5
```

### description

Данная команда используется для изменения описания конфигурируемого интерфейса.

Использование отрицательной формы команды (no) удаляет установленное описание.

#### Синтаксис

```
description <DESCRIPTION>
```

```
no description
```

#### Параметры

<DESCRIPTION> – описание интерфейса, задаётся строкой до 255 символов.

## Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-LOOPBACK

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-SERVICE-PORT

CONFIG-CELLULAR-PROFILE

CONFIG-CELLULAR-MODEM


CONFIG-VOICE-PORT

CONFIG-VOIP-PROFILE

CONFIG-LINE-AUX<sup>1</sup>

### Пример

```
esr(config-if-gi)# description "Uplink interface"
```

 <sup>1</sup> Только для ESR-21.

### history statistics

Данной командой включается запись статистики использования текущего интерфейса.

Использование отрицательной формы команды (no) отключает запись статистики использования текущего интерфейса.

### Синтаксис

[no] history statistics

### Параметры

Отсутствуют.

## Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
 CONFIG-IF-TE  
 CONFIG-IF-TWE  
 CONFIG-IF-FO  
 CONFIG-IF-HU  
 CONFIG-IF-OOB  
 CONFIG-IF-SUB  
 CONFIG-IF-QINQ  
 CONFIG-SERIAL  
 CONFIG-IF-PORT-CHANNEL  
 CONFIG-LOOPBACK  
 CONFIG-IF-MULTILINK  
 CONFIG-IF-E1  
 CONFIG-BRIDGE  
 CONFIG-CELLULAR-MODEM

### Пример

```

esr(config-if-gi)# history statistics
esr(config-if-gi)#
  
```

### interface


Данная команда позволяет перейти в режим конфигурирования одного или более интерфейсов. Использование отрицательной формы команды (no) восстанавливает настройки интерфейса по умолчанию.

### Синтаксис

```
[no] interface <IF>
```

### Параметры

<IF> – наименование интерфейса или группы интерфейсов, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

 На маршрутизаторах ESR-1000 не поддерживается работа модулей ToPGATE-WAN-E1 с аппаратной версией (hardware revision) 812.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

### Пример 1

Переход в режим конфигурирования Ethernet-интерфейса gi 1/0/20:

```
esr(config)# interface gigabitethernet 1/0/20  
esr(config-if-gi)#
```

### Пример 2

Переход в режим конфигурирования Ethernet-интерфейса te 1/0/2:

```
esr(config)# interface tengigabitethernet 1/0/2  
esr(config-if-te)#
```

### Пример 3

Переход в режим конфигурирования виртуального интерфейса:

```
esr(config)# interface loopback 5  
esr(config-loopback)#
```

### Пример 4

Переход в режим конфигурирования суб-интерфейса:

```
esr(config)# interface gigabitethernet 1/0/20.20  
esr(CONFIG-IF-SUB)#
```

### Пример 5

Переход в режим конфигурирования интерфейса port-channel 2:

```
esr(config)# interface port-channel 2  
esr(CONFIG-IF-PORT-CHANNEL)#
```

### Пример 6

Переход в режим конфигурирования интерфейса e1 1/0/1:

```
esr(config)# interface e1 1/0/1  
esr(config-if-e1)#
```



## Пример 7

Переход в режим конфигурирования интерфейса multilink 1:

```
esr(config)# interface multilink 1  
esr(CONFIG-IF-MULTILINK)#
```

### ip tcp adjust-mss

Данной командой переопределяется значение поля MSS (Maximum segment size) во входящих TCP-пакетах.

Использование отрицательной формы команды (no) отключает корректировку значение поля MSS.

### Синтаксис

```
ip tcp adjust-mss <MSS>  
no ip tcp adjust-mss
```

### Параметры

<MSS> – значение MSS, принимает значения в диапазоне [500..1460].

### Значение по умолчанию

1460

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-OOB  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-LOOPBACK

### Пример

```
esr(config-if-gi)# ip tcp adjust-mss 1400
```

## load-average

Данной командой устанавливается интервал времени, по которому происходит расчет утилизации интерфейса.

Использование отрицательной формы команды устанавливает значение по умолчанию.

### Синтаксис

```
load-average <TIME>
```

```
no load-average
```

### Параметры

<TIME> – интервал в секундах, принимает значения [5..150].

### Значение по умолчанию

5

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-CELLULAR-MODEM

### Пример

```
esr(config-if-gi)# load-average 30
```

## mode

Данной командой устанавливается режим работы физического/агрегированного интерфейса. Использование отрицательной формы команды (no) устанавливает значение режима по умолчанию.

### Синтаксис

```
mode <MODE>
no mode
```

### Параметры

<MODE> – режим работы физического/агрегированного интерфейса, принимает значения:

- switchport – устанавливает L2-режим. Возможно разрешение vlan, но запрещает назначение IP-адреса и создание саб/qinq-интерфейсов.
- routerport – устанавливает L3-режим. Возможно назначение IP-адреса и создание саб/qinq-интерфейсов, но запрещено разрешение vlan на интерфейсе.
- hybrid – возможно разрешение vlan, назначение IP-адреса и создание саб/qinq-интерфейсов<sup>1</sup>.

### Значение по умолчанию

routerport

### Необходимый уровень привилегий


10

### Командный режим

```
CONFIG-IF-GI
CONFIG-IF-TE
CONFIG-IF-TWE
CONFIG-IF-FO
CONFIG-IF-HU
CONFIG-IF-PORT-CHANNEL
```

### Пример

```
esr(config-if-gi)# mode switchport
```

 <sup>1</sup> Доступно только на маршрутизаторах ESR-1000/1200/1500/1511/1700.

## mtu

Данной командой указывается размер MTU (Maximum Transmission Unit) для интерфейсов. Использование отрицательной формы команды (no) устанавливает значение MTU по умолчанию.

## Синтаксис

```
mtu <MTU>
```

```
no mtu
```

## Параметры

<MTU> – значение MTU в байтах, принимает значения в диапазоне:

ESR-10/12V/12VF/15/15R/15VF – [552..9600],

ESR-20/21/30/31 – [552..9500],

ESR-100/200/1000/1200/1500/1511/1700 – [552..10000],

ESR-3100/3200/3200L/3300 – [552..9190].

Для Serial, E1 и multilink-интерфейсов – [552-1500] для всех моделей маршрутизаторов.

Значения MTU более 1500 можно выставлять только при включенной поддержке Jumbo-фреймов, описанной в разделе [system jumbo-frames](#).

## Значение по умолчанию

1500

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-CELLULAR-MODEM

## Пример

```
esr(config-if-gi)# mtu 1400
```

### rate-limit arp-broadcast

Данной командой включается ограничение приема arp-запросов на bridge-интерфейсе.

При использовании отрицательной формы команды (no) отключается ограничение приема arp-запросов на bridge-интерфейсе.

#### Синтаксис

```
[no] rate-limit arp-broadcast
```

#### Параметры

Отсутствуют.

#### Значение по умолчанию

Выключено.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-BRIDGE

#### Пример

```
esr(config-bridge)# rate-limit arp-broadcast
```

### rate-limit arp-broadcast pps

Данной командой устанавливается максимальный порог приема arp-запросов на bridge-интерфейсе. Данный параметр работает только при включенном режиме ограничения приема arp-запросов на bridge-интерфейсе.

При использовании отрицательной формы команды (no) устанавливает значение по умолчанию.

#### Синтаксис

```
rate-limit arp-broadcast pps <PPS>
```

```
no rate-limit arp-broadcast pps
```

#### Параметры

<PPS> – количество arp-запросов, задается в диапазоне [1..65535].

#### Значение по умолчанию

100

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-BRIDGE

**Пример**

```
esr(config-bridge)# rate-limit arp-broadcast pps 2000
```

**show interfaces counters**

Командой выполняется просмотр счетчиков на системных интерфейсах: портах, саб-интерфейсах, группах агрегации, сетевых мостах.

**Синтаксис**

```
show interfaces counters [ <IF> ]
```

**Параметры**

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

Команда отображает счётчики для портов маршрутизатора, саб-интерфейсов и туннельных интерфейсов.

Можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будут отображены счетчики всех интерфейсов заданной группы. Если задан определённый интерфейс, то будет отображена детальная информация по данному интерфейсу. При выполнении команды без параметра будут показаны счетчики всех системных интерфейсов.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример**

```
esr# show interfaces counters gigabitethernet 1/0/4-6
Interface      UC recv      Bytes recv   Errors recv   MC recv
-----
gi1/0/4        0             0             0             0
gi1/0/5        0             0             0             0
gi1/0/6        0             0             0             0
Interface      UC sent      Bytes sent   Errors sent
-----
gi1/0/4        0             0             0
gi1/0/5        1138         393748       0
```

```

gil/0/6          0          0          0
esr# show interfaces counters gigabitethernet 1/0/4
Packets received:          0
Bytes received:            0
Dropped on receive:       0
Receive errors:           0
Multicasts received:     0
Receive length errors:    0
Receive buffer overflow errors: 0
Receive CRC errors:       0
Receive frame errors:     0
Receive FIFO errors:      0
Receive missed errors:    0
Receive compressed:       0
Packets transmitted:     0
Bytes transmitted:        0
Dropped on transmit:     0
Transmit errors:          0
Transmit aborted errors:  0
Transmit carrier errors:  0
Transmit FIFO errors:     0
Transmit heartbeat errors: 0
Transmit window errors:   0
Transmit compressed:      0
Collisions:               0

```

## show interfaces description

Команда используется для просмотра описания системных интерфейсов.

### Синтаксис

```
show interfaces description [ <IF> ]
```

### Параметры

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

В команде можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будут отображены статусы всех интерфейсов заданной группы. При выполнении команды без параметра будут показаны описания всех системных интерфейсов.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
esr# show interfaces description gigabitethernet 1/0/4-5
```

Interface	Admin State	Link State	Description
-----	-----	-----	-----

```

gi1/0/4      Up      Down   Link to NSK
gi1/0/5      Up      Down   Link to MSK

```

## show interfaces history

Команда используется для просмотра статистики использования интерфейса.

### Синтаксис

```
show interfaces history [<IF>] [timer <TIMER>]
```

### Параметры

<IF> – наименование интерфейса или группы интерфейсов, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

<TIMER > – не обязательный ключ timer. В качестве параметров для данного ключа могут выступать:

- hours отображает историю за последние 72 часа,
- minutes отображает историю за последние 60 минут,
- seconds отображает историю за последние 60 секунд,
- при отсутствии ключа timer выводятся 3 таблицы истории использования туннеля(ей).

### Необходимый уровень привилегий

5

### Командный режим

ROOT

### Пример

```

esr# show interfaces history gi 1/0/1 timer minutes
gi1/0/1
Last 60 minutes:
Timer  Recv utilization, Kbit/s  Sent utilization, Kbit/s  Recv errors  Sent errors  Output
drops
-----
0-1    240          16          0          0          0
1-2    961          64          0          0          0
2-3    962          64          0          0          0
3-4    962          64          0          0          0
4-5    960          64          0          0          0
5-6    961          64          0          0          0
6-7    719          64          0          0          0
7-8    960          64          0          0          0
8-9    800          65          0          0          0
9-10   962          64          0          0          0
10-11  865          64          0          0          0
11-12  962          64          0          0          0
12-13  817          65          0          0          0
13-14  962          65          0          0          0
14-15  961          65          0          0          0
15-16  880          60          0          0          0
16-17  960          63          0          0          0

```



```

17-18 0          0          0          0          0
18-19 0          0          0          0          0
19-20 0          0          0          0          0
20-21 0          0          0          0          0
21-22 0          0          0          0          0

```

## show interfaces status

Команда используется для просмотра состояния системных интерфейсов.

### Синтаксис

```
show interfaces status [<IF>]
```

### Параметры

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

В команде можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будут отображены статусы всех интерфейсов заданной группы. Если задан конкретный интерфейс, то будет отображена детальная информация по данному интерфейсу. При выполнении команды без параметра будут показаны статусы всех системных интерфейсов.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```

esr-200# show interfaces status gigabitethernet 1/0/1-2
Interface           Admin   Link   MTU   MAC address           Last change           Mode
                   State   State  -----
-----
gil/0/1             Up     Up     1500  a8:f9:4b:af:20:f6    25,06:52:50          routerport
gil/0/2             Up     Up     1500  a8:f9:4b:af:20:f7    25,06:53:03          routerport

```

## show interfaces utilization

Команда используется для просмотра текущей нагрузки на физических интерфейсах.

### Синтаксис

```
show interfaces utilization [ <IF> ]
```

### Параметры

<IF> – имя физического интерфейса устройства, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

Можно указать несколько интерфейсов перечислением через запятую «,» либо указать диапазон интерфейсов через дефис «-». Если не указывать индексы интерфейсов, то будет отображена текущая нагрузка для всех интерфейсов заданной группы. При выполнении команды без параметра будет показана текущая нагрузка для всех физических интерфейсов.

### Необходимый уровень привилегий

1


### Командный режим

ROOT

### Пример

```
esr# show interfaces utilization gigabitethernet 1/0/3-5,1/0/9
Port          Period, s      Sent,          Recv,          Frames Sent    Frames Recv
-----
gil/0/3       5              0              0              0              0
gil/0/4       5              0              0              0              0
gil/0/5       5              0              0              0              0
gil/0/9       5              0              0              0              0
```

### show system jumbo-frames

 В текущей версии ПО данная команда поддерживается только на маршрутизаторах ESR-200/1000/1200/1500/1511/1700.

Команда используется для просмотра текущего состояния и состояния после перезагрузки устройства функции jumbo-фреймов.

### Синтаксис

```
show system jumbo-frames
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
esr# show system jumbo-frames
Jumbo frames are disabled
Jumbo frames will be disabled after reset
```

## shutdown

Данной командой отключается конфигурируемый интерфейс.

Использование отрицательной формы команды (no) включает конфигурируемый интерфейс.

### Синтаксис

```
[no] shutdown
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-LOOPBACK

CONFIG-IF-E1

CONFIG-IF-MULTILINK

### Пример

```
esr(config-if-gi)# shutdown
```

Конфигурируемый интерфейс отключен.

## speed

Данной командой устанавливается значение скорости для конфигурируемого интерфейса, группы интерфейсов. Командой могут быть установлены следующие режимы: 10 Мбит/с, 100 Мбит/с, 1000 Мбит/с, 10 Гбит/с, 25 Гбит/с, 40 Гбит/с или auto.

Использование отрицательной (no) формы команды устанавливает значение по умолчанию.

## Синтаксис

```
speed <SPEED> <DUPLEX>
```

```
no speed
```

## Параметры

<SPEED> – значение скорости:

- 10M – значение скорости 10 Мбит/с;
- 100M – значение скорости 100 Мбит/с;
- 1000M – значение скорости 1000 Мбит/с;
- 10G – значение скорости 10 Гбит/с;
- 25G – значение скорости 25 Гбит/с;
- 40G – значение скорости 40 Гбит/с;
- auto – автоматический выбор режима (недоступно для 10G-интерфейсов).

<DUPLEX> – режим работы приемопередатчика, принимает значения:

- full-duplex – дуплекс;
- half-duplex – полудуплекс.

## Значение по умолчанию

auto

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-PORT-CHANNEL

## Пример 1

```
esr(config-if-te)# speed 10G
```

Установлен скоростной режим интерфейса 10 Гбит/с.

## Пример 2

```
esr(config-if-gi)# speed 10M full-duplex
```

Установлен скоростной режим интерфейса 10 Мбит/с, дуплекс.

**switchport dot1q ethertype egress stag**

Данной командой настраивается EtherType для сервисного VLAN в исходящих пакетах. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
[no] switchport dot1q ethertype egress stag { 802.1q | 802.1ad }
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

802.1q (0x8100)

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

**Пример**

```
esr(config-if-gi)# switchport dot1q ethertype egress stag 802.1ad
```

**switchport mode**

Данная команда используется для задания режима работы интерфейса с VLAN. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
switchport mode <MODE>
```

```
no switchport mode
```

**Параметры**

<MODE> – режим работы:

- access<sup>1</sup> – интерфейс доступа, нетегированный интерфейс для одной VLAN;

- trunk<sup>1</sup> – интерфейс, принимающий только тегированный трафик за исключением одного VLAN, который может быть добавлен с помощью команды *switchport trunk native vlan*, описанной в [switchport trunk native-vlan](#);
- general<sup>2</sup> – интерфейс переключается в режим general;
- e1 – физический интерфейс переключается в режим E1 (не применима для агрегированных интерфейсов).

### Значение по умолчанию

access<sup>1</sup>

general<sup>2</sup>

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE


CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

### Пример

```
esr(config-if-gi)# switchport mode trunk
```

-  <sup>1</sup> Данная команда поддерживается только на маршрутизаторах ESR-10/12V/12VF/15/15R/15VF/20/21/30/31/100/200/3100/3200/3200L/3300.
- <sup>2</sup> Данная команда поддерживается только на маршрутизаторах ESR-1000/1200/1500/1511/1700.

### system jumbo-frames

Данной командой включается поддержка Jumbo-фреймов. Для вступления изменений в силу требуется перезагрузка устройства.

Использование отрицательной формы команды (no) выключает поддержку Jumbo-фреймов.

### Синтаксис

[no] system jumbo-frames

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# system jumbo-frames
```

## Физические интерфейсы

### snmp trap link-status

Данная команда используется для включения отправки snmp-trap о включении/выключении туннеля. Использование отрицательной формы команды (no) отключает отработку snmp-trap о включении/отключении туннеля.

### Синтаксис

```
[no] snmp trap link-status
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Включено.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO


CONFIG-IF-HU

CONFIG-IF-OOB

## Пример

```
esr(config-if-gi)# snmp trap link-status
```

## show interfaces protected-ports

 В текущей версии ПО данная команда поддерживается только на маршрутизаторах ESR-1000/1200/1500/1511/1700.

Команда используется для просмотра физических интерфейсов в режиме изоляции по группам.

### Синтаксис

```
show interfaces protected-ports [ <IF> ]
```

### Параметры

<IF> – имя физического или агрегированного интерфейса устройства, задаётся в виде, описанном в разделе [Типы и порядок именованя интерфейсов маршрутизатора](#).

В команде можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будут отображены статусы всех интерфейсов заданной группы. Если задан определенный интерфейс, то будет отображена детальная информация по данному интерфейсу. При выполнении команды без параметра будут показаны статусы всех физических интерфейсов.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
esr# show interfaces protected-ports
Interface   State           Community
-----
g11/0/5     Protected      4
```

## show interfaces sfp

Команда используется для просмотра информации об SFP-трансиверах.

### Синтаксис

```
show interfaces sfp [ <IF> ]
```

### Параметры

<IF> – имя физического интерфейса устройства, задаётся в виде, описанном в разделе [Типы и порядок именованя интерфейсов маршрутизатора](#). В команде можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будут отображены статусы всех интерфейсов заданной группы. Если задан определенный интерфейс, то будет отображена детальная информация по данному интерфейсу. При выполнении команды без параметра будут показаны статусы всех системных интерфейсов.



## Необходимый уровень привилегий

1

## Командный режим

ROOT


## Пример

```

esr# show interfaces sfp
Interface 'te1/0/1':
SFP present:      Yes
Connector Type:   LC
Type:             SFP/SFP+
Compliance code: 10G BASE-SR
Laser wavelength: 850 nm
Transfer distance: 300.00 m
Vendor OUI:       24:00:00
Vendor name:      Modultech
Vendor PN:        MT-PP-85192-SR
Vendor SN:        M1204011007
Vendor date:      04.05.12
Vendor revision:  1.0
DDM supported:    Yes
Temperature:      40.562 C
Voltage:          3.3364 V
Current:          6.004 mA
RX Power:         0.0001 mW / -40.0000 dBm
TX Power:         0.4398 mW / -3.5674 dBm
RX LOS:          Yes
TX Fault:         No
TX Disable:       No
Soft TX Disable: No
Interface 'te1/0/2':
SFP present:      Yes
Connector Type:   SC
Type:             SFP/SFP+
Compliance code: 1000BASE-LX
Laser wavelength: 1310 nm
Transfer distance: 20.00 km
Vendor OUI:       --
Vendor name:      OEM
Vendor PN:        APSB35123CXS20
Vendor SN:        SG35224701333
Vendor date:      12.12.12
Vendor revision:  1.00
DDM supported:    No

```

## show interfaces switch-port configuration

 В текущей версии ПО данная команда поддерживается только на маршрутизаторах ESR-1000/1200/1500/1500/1511/1700.

Командой выполняется просмотр параметров конфигурации физических интерфейсов.

## Синтаксис

```
show interfaces switch-port configuration [ <IF> ]
```

## Параметры

<IF> – имя физического интерфейса устройства, задаётся в виде, описанном в разделе [Типы и порядок именованя интерфейсов маршрутизатора](#).

Можно указать несколько интерфейсов перечислением через запятую либо указать диапазон интерфейсов через дефис «-». Если не указывать индексы интерфейсов, то будут отображены параметры всех интерфейсов заданной группы. При выполнении команды без параметра будут показаны параметры всех физических интерфейсов.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
esr# show interfaces switch-port configuration gigabitethernet 1/0/5-7
```

Port	Media	Duplex	Speed	Neg	Flow control	Admin State	Back Pressure
-----	-----	-----	-----	-----	-----	-----	-----
gi1/0/5	none	Half	10 Mbps	Enabled	Off	Up	Disabled
gi1/0/6	none	Half	10 Mbps	Enabled	Off	Up	Disabled
gi1/0/7	none	Half	10 Mbps	Enabled	Off	Up	Disabled

## show interfaces switch-port status

Команда используется для просмотра состояния физических интерфейсов.

## Синтаксис

```
show interfaces switch-port status [ <IF> ]
```

## Параметры

<IF> – имя физического или агрегированного интерфейса устройства, задаётся в виде, описанном в разделе [Типы и порядок именованя интерфейсов маршрутизатора](#).

Можно указать несколько интерфейсов перечислением через запятую «,» либо указать диапазон интерфейсов через дефис «-». Если не указывать индексы интерфейсов, то будут отображены статусы всех интерфейсов заданной группы. Если задан определенный интерфейс, то будет отображена детальная информация по данному интерфейсу. При выполнении команды без параметра будут показаны статусы всех физических интерфейсов.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

### Пример

```

esr# show interfaces switch-port status
Port          Media    Duplex  Speed  Neg      Flow  Link  Back  MDI  Port
-----     -
gii/0/1      --      --      --     Enabled --    Down  --    --   access
gii/0/2      --      --      --     Enabled --    Down  --    --   access
gii/0/3      --      --      --     Enabled --    Down  --    --   access
gii/0/4      --      --      --     Enabled --    Down  --    --   access
gii/0/5      --      --      --     Enabled --    Down  --    --   access
gii/0/6      --      --      --     Enabled --    Down  --    --   access

esr# show interfaces switch-port status gigabitethernet 1/0/2
Interface      gigabitethernet 1/0/2
Status:        up
Media:         copper
Speed:         100 Mbps
Duplex:        full
Flow control:  no
MAC address:   a8:f9:b5:00:00:25
MAC status:
  Buffers full:      no
  Doing back pressure: no
  Sending PAUSE frames: no
  Receiving PAUSE frames: no
  Auto-Negotiation done: yes
  Sync fail:         no

```

### threshold rx-utilization

Данная команда используется для задания порога входящей нагрузки интерфейса для отправки snmp-trap eltexInterfaceRxUtilizationHigh и eltexInterfaceRxUtilizationHighOk.

Использование отрицательной формы команды (no) устанавливает значения по умолчанию.

### Синтаксис

```

threshold rx-utilization <TH-HIGH> clear <TH-LOW>
no threshold rx-utilization

```

### Параметры

<TH-HIGH> – порог в процентах для отправки snmp-trap eltexInterfaceRxUtilizationHigh;

<TH-LOW> – порог в процентах для отправки snmp-trap eltexInterfaceRxUtilizationHighOk.

### Значение по умолчанию

<TH-HIGH> – 90%;

<TH-LOW> – 85%.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

**Пример**

```
esr(config-if-gi)# threshold rx-utilization 90 clear 80
```

**threshold rx-utilization monitoring**

Данная команда используется для включения отправки snmp-trap eltexInterfaceRxUtilizationHigh и eltexInterfaceRxUtilizationHighOk.

Использование отрицательной формы команды (no) отключает отправки snmp-trap eltexInterfaceRxUtilizationHigh и eltexInterfaceRxUtilizationHighOk.

**Синтаксис**

```
[no] threshold rx-utilization monitoring
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

## Пример

```
esr(config-if-gi)# threshold rx-utilization monitoring
```

## threshold tx-utilization

Данная команда используется для задания порогов исходящей нагрузки интерфейса для отправки snmp-trap eltexInterface TxUtilizationHigh и eltexInterface TxUtilizationHighOk.

Использование отрицательной формы команды (no) устанавливает значения по умолчанию.

## Синтаксис

```
threshold tx-utilization <TH-HIGH> clear <TH-LOW>
```

```
no threshold tx-utilization
```

## Параметры

<TH-HIGH> – порог в процентах для отправки snmp-trap eltexInterfaceTxUtilizationHigh;

<TH-LOW> – порог в процентах для отправки snmp-trap eltexInterfaceTxUtilizationHighOk.

## Значение по умолчанию

<TH-HIGH> – 90%;

<TH-LOW> – 85%.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

## Пример

```
esr(config-if-gi)# threshold tx-utilization 90 clear 80
```

## threshold tx-utilization monitoring

Данная команда используется для включения отправки snmp-trap eltexInterface TxUtilizationHigh и eltexInterface TxUtilizationHighOk.

Использование отрицательной формы команды (no) отключает отправку snmp-trap eltexInterface T xUtilizationHigh и eltexInterface T xUtilizationHighOk.

### Синтаксис

```
[no] threshold tx-utilization monitoring
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

### Пример

```
esr(config-if-gi)# threshold tx-utilization monitoring
```

### switchport community

Данной командой интерфейс добавляется в группу изоляции. Данная команда актуальна, только если порт находится в режиме изоляции по группам.

Использование отрицательной формы команды (no) удаляет интерфейс из группы изоляции.

### Синтаксис

```
switchport community <ID>
```

```
no switchport community
```

### Параметры

<ID> – идентификатор группы, принимает значения в диапазоне [1..30].

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-F0

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

## Пример

```
esr(config-if-gi)# switchport community 10
```

## switchport protected

Данной командой на интерфейсе включается функция Private VLAN и указывается интерфейс, на который могут отправляться принятые пакеты.

Использование отрицательной формы команды (no) отключает функцию Private VLAN.

## Синтаксис

```
switchport protected <IF>
```

```
no switchport protected
```

## Параметры

<IF> – наименование интерфейса, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

## Значение по умолчанию

Функция Private VLAN отключена.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-F0

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

## Пример

```
esr(config-if-gi)# switchport protected gigabitethernet 1/0/1
```

### switchport protected-port

Данной командой интерфейс переводится в режим изоляции по группам. В данном режиме обмен трафиком между интерфейсами одной группы разрешен, обмен трафиком между интерфейсами разных групп запрещен, обмен трафиком между изолированными и неизолированными интерфейсами разрешен.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
[no] switchport protected-port
```

### Параметры

Отсутствуют.

### Значение по умолчанию

Интерфейс не изолирован.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

## Пример

```
esr:esr(config-if-gi)# switchport protected-port
```

## Агрегированные интерфейсы

### channel-group

Данной командой физический интерфейс включается в группу агрегации каналов.

Использование отрицательной формы команды (no) удаляет интерфейс из группы агрегации каналов.



**Синтаксис**

```
channel-group <ID> mode <MODE>
no channel-group
```

**Параметры**

<ID> – порядковый номер группы агрегации каналов, принимает значения [1..12].

<MODE> – режим формирования группы агрегации каналов:

- auto – добавить интерфейс в динамическую группу агрегации с поддержкой протокола LACP;
- on – добавить интерфейс в статическую группу агрегации.

**Необходимый уровень привилегий**

10

**Командный режим**

```
CONFIG-IF-GI
CONFIG-IF-TE
CONFIG-IF-TWE
CONFIG-IF-FO
CONFIG-IF-HU
```

**Пример**

```
esr(config-if-gi)# channel-group 6 mode auto
```

**lasp port-priority**

Данной командой устанавливается LACP-приоритет интерфейса Ethernet.

Использование отрицательной формы команды устанавливает значение по умолчанию.

**Синтаксис**

```
lasp port-priority <PRIORITY>
no lasp port-priority
```

**Параметры**

<PRIORITY> – приоритет, указывается в диапазоне [1..65535].

**Значение по умолчанию**

1

**Необходимый уровень привилегий**

10

## Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU

## Пример

```
esr(config-if-gi)# lacp port-priority 5000
```

## lacp system-priority

Данной командой устанавливается приоритет системы для протокола LACP.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
lacp system-priority <PRIORITY>  
no lacp system-priority
```

## Параметры

<PRIORITY> – приоритет, указывается в диапазоне [1..65535].

## Значение по умолчанию

1

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# lacp system-priority 5000
```

## lacp timeout

Данной командой устанавливается административный таймаут протокола LACP.

Использование отрицательной формы команды устанавливает значение по умолчанию.

**Синтаксис**

```
lACP timeout { short | long }
no lACP timeout
```

**Параметры**

long – длительное время таймаута (90 секунд);  
short – короткое время таймаута (3 секунды).

**Значение по умолчанию**

long

**Необходимый уровень привилегий**

10

**Командный режим**

```
CONFIG-IF-GI
CONFIG-IF-TE
CONFIG-IF-TWE
CONFIG-IF-FO
CONFIG-IF-HU
```

**Пример**

```
esr(config-if-gi)# lACP timeout short
```

**port-channel load-balance**

Данной командой устанавливается механизм балансировки нагрузки для групп агрегации каналов. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
port-channel load-balance {src-dst-mac-ip|src-dst-mac|src-dst-ip|src-dst-mac-ip-port}
no port-channel load-balance
```

**Параметры для ESR-1000, ESR-1200, ESR-1500, ESR-1511 и ESR-1700**

src-dst-mac – механизм балансировки основывается на MAC-адресе отправителя и получателя;

src-dst-ip – механизм балансировки основывается на IP-адресе отправителя и получателя;

src-dst-mac-ip – механизм балансировки основывается на MAC-адресе и IP-адресе отправителя и получателя;

src-dst-ip-port – механизм балансировки основывается на IP-адресах и tcp/udp-портах отправителя и получателя;

src-dst-mac-ip-port – механизм балансировки основывается на MAC-адресе, IP-адресе и порте отправителя и получателя.

**Параметры для ESR-10, ESR-12V, ESR-12VF, ESR-15, ESR-15R, ESR-15VF, ESR-20, ESR-21, ESR-30, ESR-31, ESR-100, ESR-3100, ESR-3200, ESR-3200L, ESR-3300**

active-backup – весь трафик отправляется в один интерфейс и перенаправляется в другой, если первый перешел в состояние down;

src-dst-mac – механизм балансировки основывается на MAC-адресах отправителя и получателя;

src-dst-mac-ip – механизм балансировки основывается на MAC-адресах и IP-адресах отправителя и получателя;

src-dst-ip-port – механизм балансировки основывается на IP-адресах и tcp/udp-портах отправителя и получателя.

**Значение по умолчанию**

src-dst-mac

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# port-channel load-balance src-dst-mac-ip
```

**show interfaces port-channel**

Данная команда используется для просмотра информации о членах группы агрегации каналов.

**Синтаксис**

```
show interfaces port-channel [<ID>]
```

**Параметры**

<ID> – порядковый номер группы агрегации каналов, принимает значения в диапазоне [1..12].

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

## Пример

```

esr# show interfaces port-channel 1
load-balance: src-dst-mac
Channels    Ports
-----
po1         gi1/0/21

```

## show lacp counters

**⚠** В текущей версии ПО данный функционал поддерживается только на маршрутизаторах ESR-1000/1200/1500/1511/1700.

Данная команда используется для просмотра статистики работы LACP-протокола для интерфейса Ethernet.

## Синтаксис

```
show lacp counters [ <IF> ]
```

## Параметры

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

В команде можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будет отображена статистика всех интерфейсов заданной группы. Если задан определенный интерфейс, то будет отображена детальная информация по данному интерфейсу.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```

esr# show lacp counters port-channel 2
Interface      Sent      Recv      Link failure
-----
po2             42       814       2

```

## show lacp interfaces

Данная команда используется для просмотра информации о протоколе LACP.

## Синтаксис

```
show lacp interfaces [ <IF> ]
```

## Параметры

<IF> – имя физического или агрегированного интерфейса устройства, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

В команде можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будет отображена информация о LACP-протоколе для всех интерфейсов заданной группы.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```

esr# show lacp interfaces port-channel 2
port-channel 2 [aggregator 1, active] ports count: 1
-----
Actor Port                Partner Port
-----
System Priority            32768                1
System MAC                 a8:f9:4b:aa:12:40    a8:f9:4b:83:01:80
Key                        8000                 1
port-channel 2 [aggregator 2, backup] ports count: 1
-----
Actor Port                Partner Port
-----
System Priority            32768                65535
System MAC                 a8:f9:4b:aa:12:40    00:00:00:00:00:00
Key                        8000                 FFFF
esr# show lacp interfaces gigabitethernet 1/0/1
gigabitethernet 1/0/1 [active] up
-----
Actor Port                Partner Port
-----
Port Priority              32768                1
LACP Activity              Active                Active

```

## show lacp parameters

Данная команда используется для просмотра параметров настройки протокола LACP для интерфейса Ethernet.

## Синтаксис

```
show lacp parameters [ <IF> ]
```

## Параметры

<IF> – имя физического интерфейса устройства, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#). Возможно использование только физических интерфейсов.

В команде можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будут отображены параметры всех интерфейсов заданной группы. Если задан конкретный интерфейс, то будет отображена детальная информация по данному интерфейсу.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример**

```

esr# show lacp parameters tengigabitethernet 1/0/2
  LACP parameters
  ~~~~~
Interface   Port Priority   Timeout   Mode
-----
te1/0/2     32768          Short     Active

```

**E1/multilink****enable**

Данной командой включается PPP-пользователь.

Использование отрицательной формы команды (no) отключает PPP-пользователя.

**Синтаксис**

```
[no] enable
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

PPP-пользователь отключен.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-PPP-USER

**Пример**

```
esr(config-ppp-user)# enable
```

## ip tcp header-compression

Данной командой включается протокол сжатия tcp-заголовком. Протокол используется для улучшения производительности низкоскоростных каналов связи.

Использование отрицательной формы команды (no) отключает протокол сжатия tcp-заголовков.

### Синтаксис

```
[no] ip tcp header-compression
```

### Параметры

Отсутствуют.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-E1

CONFIG-IF-MULTILINK

### Пример

```
(config-if-e1)# ip tcp header-compression
```

## ip tcp compression-connections

Данной командой указывается количество одновременных tcp-соединений, для которых будет использоваться протокол компрессии tcp-заголовков.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию для количества одновременных tcp-соединений, для которых будет использоваться протокол компрессии tcp-заголовков.

### Синтаксис

```
ip tcp compression-connections <NUMBER>
```

```
no ip tcp compression-connections
```

### Параметры

<NUMBER> – количество одновременных tcp-соединений, для которых будет использоваться протокол компрессии tcp-заголовков. Может принимать значения [2..16].

### Значение по умолчанию

16



**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-E1

CONFIG-IF-MULTILINK

**Пример**

```
esr:esr(config-if-e1)# ip tcp compression-connections 32
esr:esr(config-if-e1)#
```

**mrru**

Данная команда определяет максимальный размер принимаемого пакета для MLPPP-интерфейса. Использование отрицательной формы команды (no) устанавливает значение mrru по умолчанию.

**Синтаксис**

```
mrru { <MRRU> }
```

```
no mrru
```

**Параметры**

<MRRU> – максимальный размер принимаемого пакета для MLPPP-интерфейса, принимает значения в диапазоне [1500..10000].

**Значение по умолчанию**

1500

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-MULTILINK

**Пример**

```
esr(CONFIG-IF-MULTILINK)# mrru 1700
```

**password**

Команда для установки пароля в открытой или зашифрованной форме определенному пользователю для аутентификации удаленной стороны. Пароль пользователя хранится в конфигурации в

зашифрованной форме. При конфигурировании можно задать пароль в открытой форме либо скопировать пароль в зашифрованной форме с другого устройства.

Использование отрицательной формы команды (no) удаляет пароль пользователя.


### Синтаксис

```
password ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no password
```

### Параметры

<CLEAR-TEXT> – пароль в открытой форме, задаётся строкой [1..64] символов, может включать символы [0-9a-fA-F];

<ENCRYPTED-TEXT> – пароль в зашифрованной форме, задаётся строкой [2..128] символов.

 Пароли хранятся в конфигурации в зашифрованной форме независимо от формата, использованного при вводе команды.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-PPP-USER

### Пример

```
esr(config-ppp-user)# password ascii-text 01234567
```

### ppp authentication chap

Данной командой включается CHAP-аутентификация.

Использование отрицательной формы команды (no) отключает аутентификацию.

### Синтаксис

```
[no] ppp authentication chap
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-SERIAL

CONFIG-IF-E1

## CONFIG-IF-MULTILINK

### Пример

```
esr(config-if-e1)# ppp authentication chap
```

### ppp chap hostname

Данной командой указывается имя маршрутизатора, которое отправляется удаленной стороне для прохождения CHAP-аутентификации.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ppp chap hostname <NAME>
```

```
no ppp chap hostname
```

### Параметры

<NAME> – имя маршрутизатора, задаётся строкой до 31 символа.

### Значение по умолчанию

Системное имя устройства (hostname).

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-SERIAL

CONFIG-IF-E1

CONFIG-IF-MULTILINK

### Пример

```
esr(config-if-e1)# ppp chap hostname esr1
```

### ppp chap password

Данной командой указывается пароль, который отправляется удаленной стороне вместе с именем маршрутизатора для прохождения CHAP-аутентификации.

Использование отрицательной формы команды (no) удаляет пароль.

### Синтаксис


```
ppp chap password ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
```

```
no ppp chap password
```

## Параметры

<CLEAR-TEXT> – пароль в открытой форме, задаётся строкой [1..64] символов, может включать символы [0-9a-fA-F];

<ENCRYPTED-TEXT> – пароль в зашифрованной форме, задаётся строкой [2..128] символов.

 Пароль хранится в конфигурации в зашифрованной форме независимо от формата, использованного при вводе команды.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-SERIAL

CONFIG-IF-E1

CONFIG-IF-MULTILINK

## Пример

```
esr(config-if-e1)# ppp chap password ascii-text 01234567
```

## ppp chap refuse

Данной командой включается игнорирование аутентификации.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

## Синтаксис

[no] ppp chap refuse

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Игнорирование аутентификации выключено.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-SERIAL

CONFIG-IF-E1

CONFIG-IF-MULTILINK

## Пример

```
esr(config-if-e1)# ppp chap refuse
```

## ppp chap username

Данной командой указывается пользователь для аутентификации удаленной стороны и осуществляется переход в режим конфигурирования пользователя.

Использование отрицательной формы команды (no) удаляет указанного пользователя.

## Синтаксис

```
[no] ppp chap username <NAME>
```

## Параметры

<NAME> – имя пользователя, задаётся строкой до 31 символа.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-SERIAL

CONFIG-IF-E1

CONFIG-IF-MULTILINK

## Пример

```
esr(config-if-e1)# ppp chap username xap
```

## ppp ipcp accept-address

Данной командой разрешается принимать от соседа любой ненулевой IP-адрес в качестве локального IP-адреса.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
[no] ppp ipcp accept-address
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Прием IP-адреса запрещен.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-SERIAL

CONFIG-IF-E1

CONFIG-IF-MULTILINK

**Пример**

```
esr(config-if-e1)# ppp ipcp accept-address
```

**ppp ipcp remote-address**

Данной командой устанавливается IP-адрес, который отправляется удаленной стороне для последующего его присвоения.

Использование отрицательной формы команды (no) удаляет IP-адрес удаленной стороны.

**Синтаксис**

```
ppp ipcp remote-address <ADDR>
```

```
no ppp ipcp remote-address
```

**Параметры**

<ADDR> – IP-адрес удаленного шлюза.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-SERIAL

CONFIG-IF-E1

CONFIG-IF-MULTILINK

**Пример**

```
esr(config-if-e1)# ppp ipcp remote-address 192.168.1.2
```

**ppp max-configure**

Данной командой устанавливается количество попыток отправки Configure-Request пакетов, прежде чем удаленный пир будет признан неспособным ответить.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
ppp max-configure <VALUE>  
no ppp max-configure
```

**Параметры**

<VALUE> – время в секундах, принимает значения [1..255].

**Значение по умолчанию**

10

**Необходимый уровень привилегий**

10

**Командный режим**

```
CONFIG-SERIAL  
CONFIG-IF-E1  
CONFIG-IF-MULTILINK
```

**Пример**

```
esr(config-if-e1)#i ppp max-configure 4
```

**ppp max-failure**

Данной командой устанавливается количество попыток выслать Configure-NAK пакеты, прежде чем будут подтверждены все опции.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
ppp max-failure <VALUE>  
no ppp max-failure
```

**Параметры**

<VALUE> – время в секундах, принимает значения [1..255].

**Значение по умолчанию**

10

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-SERIAL

CONFIG-IF-E1

CONFIG-IF-MULTILINK

**Пример**

```
esr(config-if-e1)#ppp max-failure 3
```

**ppp max-terminate**

Данной командой устанавливается количество попыток выслать Terminate-Request пакеты, прежде чем сессия будет прервана.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
ppp max-terminate <VALUE>
```

```
no ppp max-terminate
```

**Параметры**

<VALUE> – время в секундах, принимает значения [1..255].

**Значение по умолчанию**

2

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-SERIAL

CONFIG-IF-E1

CONFIG-IF-MULTILINK

**Пример**

```
esr(config-if-e1)# ppp max-terminate 4
```

**ppp mru**

Данной командой указывается размер MRU (Maximum Receive Unit) для интерфейса.

Использование отрицательной формы команды (no) устанавливает значение MRU по умолчанию.



**Синтаксис**

```
ppp mru <MRU>
```

```
no ppp mru
```

**Параметры**

<MRU> – значение MRU, принимает значения в диапазоне [552..1500].

**Значение по умолчанию**

1500

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-SERIAL

CONFIG-IF-E1

CONFIG-IF-MULTILINK

**Пример**

```
esr(config-if-e1)# mru 1400
```

**ppp multilink**

Данной командой включается режим MLPPP на E1-интерфейсе.

Использование отрицательной формы команды (no) отключает режим MLPPP.

**Синтаксис**

```
[no] ppp multilink
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-E1

## Пример

```
esr(config-if-e1)# ppp multilink
```

### ppp multilink-group

Данной командой E1-интерфейс включается в группу агрегации.

Использование отрицательной формы команды (no) исключает интерфейс из группы агрегации.

### Синтаксис

```
ppp multilink-group <GROUP-ID>
```

### Параметры

<GROUP-ID> – идентификатор группы, принимает значение [1..4].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-E1

## Пример

```
esr(config-if-e1)# ppp multilink-group 1
```

### ppp timeout keepalive

Данной командой устанавливается интервал времени в секундах, по истечении которого маршрутизатор отправляет keepalive-сообщение.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ppp timeout keepalive [ <TIME >]
```

```
no ppp timeout keepalive
```

### Параметры

<TIME> – время в секундах, принимает значения [1..32767].

### Значение по умолчанию

10

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-SERIAL

CONFIG-IF-E1

CONFIG-IF-MULTILINK

**Пример**

```
esr(config-if-e1)# ppp timeout keepalive 200
```

**ppp timeout retry**

Данной командой устанавливается интервал времени в секундах, по истечении которого маршрутизатор повторяет запрос на установление сессии.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
ppp timeout retry <TIME>
```

```
no ppp timeout retry
```

**Параметры**

<TIME> – время в секундах, принимает значения [1..255].

**Значение по умолчанию**

3

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-SERIAL

CONFIG-IF-E1

CONFIG-IF-MULTILINK

**Пример**

```
esr(config-if-e1)# ppp timeout retry 3
```

## show controllers e1

Данной командой выводится информация о E1-контролерах.

### Синтаксис

```
show controllers e1 [<IF>]
```

### Параметры

<IF> – имя физического интерфейса маршрутизатора, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```

esr# show controllers e1
Interface 'te1/0/1':
  SFP present:      Yes
  SFP Vendor name:  NSC-COM
  SFP Vendor PN:    611.900
  Line code:        HDB3
  Clock source:     Internal
  Timeslot:         24
  Invert Data:      No
  Framing CRC4:     No
  Loopback:         --
  CRC algorithm:    FCS16
  E1 Link:          Down
  E1 Synced:        No
  E1 RX AIS:        No
  E1 RX RAI:        No

```

## switchport e1 slot

Данной командой порт e1 привязывается к физическому интерфейсу. Использование отрицательной формы команды (no) переходит в стандартный режим.

### Синтаксис

```
[no] switchport e1 slot <SLOT>
```

### Параметры

<SLOT> – идентификатор слота, принимает значение в диапазоне [0..12].

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

**Пример**

```
esr(config-if-gi)# switchport e1 0
```

**switchport e1 clock source**

Данной командой определяется источник синхронизации e1-интерфейса.

Использование отрицательной формы командой (no) устанавливает значение по умолчанию.

**Синтаксис**

```
switchport e1 clock source { internal | line }
```

```
no switchport e1 clock source
```

**Параметры**

internal – используется внутренний источник синхронизации;

line – в качестве источника синхронизации используется сигнал с линии.

**Значение по умолчанию**

internal

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

## Пример

```
esr:esr(config-if-gi)# switchport e1 clock source line
```

### switchport e1 crc

Данной командой определяется режим проверки целостности передаваемых данных. Использование отрицательной формы командой (no) устанавливает значение по умолчанию.

### Синтаксис

```
switchport e1 crc { 16 | 32 }  
no switchport e1 crc
```

### Параметры

16 – используется 16-битный алгоритм проверки отсутствия ошибок (CRC);  
32 – используется 32-битный алгоритм проверки отсутствия ошибок (CRC).

### Значение по умолчанию

16

### Необходимый уровень привилегий

10

### Командный режим

```
CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU
```

## Пример

```
esr:esr(config-if-gi)# switchport e1 crc 16
```

### switchport e1 framing

Данной командой включается проверка целостности по алгоритму CRC4. Использование отрицательной формы командой (no) устанавливает значение по умолчанию.

### Синтаксис

```
switchport e1 framing { crc4 | no-crc4 }  
no switchport e1 framing
```

## Параметры

crc4 – включение проверки целостности по CRC4;

no-crc4 – выключение проверки целостности по CRC4.

## Значение по умолчанию

Выключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

## Пример

```
esr:esr(config-if-gi)# switchport e1 framing crc4
```

## switchport e1 invert data

Данной командой включается реверсивная отправка данных.

Использование отрицательной формы командой (no) устанавливает значение по умолчанию.

## Синтаксис

```
[no] switchport e1 invert data
```

## Параметры

Отсутствуют.

## Значение по умолчанию

Выключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

### Пример

```
esr:esr(config-if-gi)# switchport e1 invert data
```

### switchport e1 linecode

Данной командой определяется алгоритм кодирования данных, осуществляемое для их передачи по физическому каналу.

Использование отрицательной формы командой (no) устанавливает значение по умолчанию.

### Синтаксис

```
switchport e1 linecode { ami | hdb3 }
```

### Параметры

ami – использовать алгоритм ami;

hdb3 – использовать алгоритм hdb3.

### Значение по умолчанию

hdb3

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

### Пример

```
esr:esr(config-if-gi)# switchport e1 linecode ami
```

### switchport e1 timeslots

Данной командой определяется количество используемых 64 кбит/с каналов в потоке E1.

Использование отрицательной формы командой (no) устанавливает значение по умолчанию.



**Синтаксис**

```
switchport e1 timeslots < RANGE >  
no switchport e1 timeslots
```

**Параметры**

< RANGE > – количество 64 кбит/с каналов, принимает значение [1..31].

**Значение по умолчанию**

31

**Необходимый уровень привилегий**

10

**Командный режим**

```
CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU
```

**Пример**

```
esr:esr(config-if-gi)# switchport e1 timeslots 16
```

**switchport e1 unframed**

Данной командой включается режим использования потока E1 как единого без разделения на каналы по 64 кбит/с.

Использование отрицательной формы командой (no) устанавливает значение по умолчанию.

**Синтаксис**

```
[no] switchport e1 unframed
```

**Параметры**

Отсутствуют.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU

**Пример**

```
esr:esr(config-if-gi)# switchport e1 unframed
```

**switchport mode e1**

Данной командой физический порт переводится в режим работы с SFP e1 модулем.  
Использование отрицательной формы команды (no) переводит порт в стандартный режим.

**Синтаксис**

```
[no] switchport mode e1
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

10


**Командный режим**

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU

**Пример**

```
esr(config-if-gi)# switchport mode e1
```

## Последовательные интерфейсы

 Только для ESR-21.

### chat-script

Команда используется для создания chat-script, которые могут использоваться при подключении через dialup.

Использование отрицательной формы команды (no) удаляет chat-script.

### Синтаксис

```
chat-script <NAME> <TEXT>
no chat-script { <NAME> | all }
```

### Параметры

<NAME> – название скрипта, задается строкой до 31 символа.

<TEXT> – содержание скрипта, задается строкой до 255 символов. Необходимо заключить скрипт в круглые скобки и экранировать при помощи символа "\" пробелы и специализированные символы.

all – ключ для удаления всех созданных скриптов.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

### Пример

```
esr(config)# chat-script DIAL "\(\ABORT ERROR ABORT BUSY \"\" \"ATZ\" OK \"ATD\" TIMEOUT 30
CONNECT\)"
```

### clear line aux

Команда используется для удаления сессий последовательных интерфейсов.

### Синтаксис

```
clear line aux [ <UNIT>/<SLOT>/<PORT> ]
```

### Параметры

<UNIT> – номер устройства в группе устройств;

<SLOT> – номер модуля в составе устройства или 0 при отсутствии деления устройства на модули;

<PORT> – порядковый номер порта.

**Необходимый уровень привилегий**

15

**Командный режим**

ROOT

**Пример**

```
esr# clear line aux 1/0/1
```

**databits**

Данной командой задается количество бит данных в посылке.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
databits <BITS>
```

```
no databits
```

**Параметры**

<BITS> – количество бит данных в посылке. Принимает значение [7..8].

**Значение по умолчанию**

8

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-LINE-AUX

**Пример**

```
esr(config-line-aux)# databits 7
```

**dialer**

Данной командой включается функция дозвонщика.

Использование отрицательной формы команды (no) отключает функцию дозвонщика.

## Синтаксис

```
[no] dialer
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Отключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-SERIAL

## Пример

```
esr(config-serial)# dialer
```

## dialer idle-timeout

Данной командой устанавливается время удержания подключения при отсутствии трафика. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
dialer idle-timeout <TIME>  
no dialer idle-timeout
```

## Параметры

<TIME> – время удержания подключения при отсутствии трафика в секундах, принимает значение в диапазоне [1..65535].

## Значение по умолчанию

30

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-SERIAL

## Пример

```
esr(config-serial)# dialer idle-timeout 210
```

## dialer in-band

Данной командой на serial-интерфейсе включается режим V.25bis.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
[no] dialer in-band
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Отключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-SERIAL

## Пример

```
esr(config-serial)# dialer in-band
```

## dialer map

Данной командой задается соответствие IP-подсети и номеров дозвона для доступа к этим подсетям.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
dialer map ip <SUBNET> name <PPPUSER> [ modem-script <SCRIPT-NAME> ] <PHONE-NUM>
no dialer map ip
```

## Параметры

<SUBNET> – адрес назначения, задается в виде AAA.BBB.CCC.DDD/NN, где AAA-DDD принимают значения [0..255] и NN принимает значения [1..32];

<PPPUSER> – имя пользователя для дозвона, задается строкой до 31 символа;

<SCRIPT-NAME> – имя скрипта дозвона, задается строкой до 31 символа;

<PHONE-NUM> – номер для дозвона. Задается строкой до 15 символов, возможно использовать только цифры.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-SERIAL

### Пример

```
esr(config-serial)# dialer map ip 192.168.33.0/27 name SITE12 5512
```

### dialer string

Данной командой задается номер телефона для установки соединения через dialup-модем.

Использование отрицательной формы команды (no) удаляет номер телефона для установки соединения через dialup-модем.

### Синтаксис

```
dialer string <PHONE-NUM>  
no dialer string
```

### Параметры

<PHONE-NUM> – номер телефона для дозвона. Задается строкой до 15 символов, возможно использовать только цифры.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-SERIAL

### Пример

```
esr(config-serial)# dialer string 3835401
```

**flowcontrol**

Данной командой задается режим управления потоком данных.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
flowcontrol <MODE>
no flowcontrol
```

**Параметры**

<MODE> – режим управления потоком. Принимает значения:

- software – программное управление потоком;
- hardware – аппаратное управление потоком;
- disabled – управление потоком отключено.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-LINE-AUX

**Пример**

```
esr(config-line-aux)# flowcontrol software
```

**line aux**

Данной командой осуществляется переход в режим конфигурирования последовательного интерфейса.

Использование отрицательной формы команды (no) удаляет последовательный интерфейс.

**Синтаксис**

```
[no] line aux [ <UNIT>/<SLOT>/<PORT> ]
```

**Параметры**

<UNIT> – номер устройства в группе устройств;

<SLOT> – номер модуля в составе устройства или 0 при отсутствии деления устройства на модули;

<PORT> – порядковый номер порта.

**Необходимый уровень привилегий**

15



## Командный режим

CONFIG

### Пример

```
esr(config)# line aux 1/0/1
esr(config-line-aux)#
```

### modem inout

Данной командой переводит последовательный интерфейс в режим работы с модемом.

Использование отрицательной формы команды (no) переводит последовательный интерфейс в режим консольного сервера.

### Синтаксис

```
modem inout
no modem inout
```

### Параметры

Отсутствуют.

### Значение по умолчанию

По умолчанию последовательный интерфейс работает в режиме консольного сервера.

### Необходимый уровень привилегий

15

## Командный режим

CONFIG-LINE-AUX

### Пример

```
esr(config-line-aux)# modem inout
```

### parity

Данной командой задается режим установки бита четности.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
parity <MODE>
no parity
```

## Параметры

<MODE> – режим установки бита четности. Принимает значения:

- odd – проверка на нечетность;
- even – проверка на четность;
- none – бит четности не выставляется.

## Значение по умолчанию

none.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-LINE-AUX

## Пример

```
esr(config-line-aux)# parity even
```

## show line aux

Команда используется для просмотра состояния последовательных интерфейсов.

## Синтаксис

```
show line aux [ <UNIT>/<SLOT>/<PORT> ]
```

## Параметры

<UNIT> – номер устройства в группе устройств;

<SLOT> – номер модуля в составе устройства или 0 при отсутствии деления устройства на модули;

<PORT> – порядковый номер порта.

## Необходимый уровень привилегий

10

## Командный режим

ROOT

## Пример

```
esr# show line aux
AUX Line 1/0/1
  Baud rate:          115200
  Databits:           8
  Parity:              NONE
```

```
Stopbits:          1
Flowcontrol:       Disabled
Timeout Exec (min): 300
Telnet port:       2001
Modem mode:        Disabled
```

## speed

Данной командой задается скорость работы последовательного интерфейса.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
speed <SPEED>
```

```
no speed
```

### Параметры

<SPEED> – скорость работы последовательного интерфейса в бит/с. Принимает значения:

- 300;
- 1200;
- 2400;
- 4800;
- 9600;
- 19200;
- 38400;
- 57600;
- 115200.

### Значение по умолчанию

115200.

### Необходимый уровень привилегий

15

Командный режим

CONFIG-LINE-AUX

### Пример

```
esr(config-line-aux)# speed 9600
```

## stopbits

Данной командой задается количество стоповых битов в посылке.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
stopbits <STOP-BITS>
```

```
no stopbits
```

### Параметры

<STOP-BITS> – количество стоповых битов в посылке. Принимает значения [1..2].

### Значение по умолчанию

1

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-LINE-AUX

### Пример

```
esr(config-line-aux)# stopbits 2
```

### transport telnet port

Данной командой задается номер TCP-порта для режима консольного сервера. При telnet-подключении на IP-адрес маршрутизатора и сконфигурированный данной командой TCP-порт пользователь подключится к консольному интерфейсу устройства, подключенного к последовательному интерфейсу маршрутизатора.

Использование отрицательной формы команды (no) удаляет значение номера TCP-порта для режима консольного сервера.

### Синтаксис

```
transport telnet port <PORT>
```

```
no transport telnet port
```

### Параметры

<PORT> – номер TCP-порта для режима консольного сервера. Принимает значения [1..65535].

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-LINE-AUX

## Пример

```
esr(config-line-aux)# transport telnet port 2001
```

## Беспроводные модемы

✘ На маршрутизаторах ESR-1700 работа беспроводных модемов не поддерживается.

### allowed-auth

Данной командой задается метод аутентификации пользователя в мобильной сети.

Использование отрицательной формы команды (no) устанавливает метод аутентификации по умолчанию.

### Синтаксис

```
allowed-auth <TYPE>
```

```
no allowed-auth
```

### Параметры

<TYPE> – метод аутентификации пользователя в мобильной сети [none, PAP, CHAP, MSCHAP, MSCHAPv2, EAP].

### Значение по умолчанию

PAP

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-CELLULAR-PROFILE

## Пример

```
esr(config-cellular-profile)# allowed-auth MSCHAP
```

### allowed-mode

Данной командой разрешается использование режима при работе USB-модема.

Использование отрицательной формы команды (no) удаляет разрешение на использование режима.

### Синтаксис

```
[no] allowed-mode <MODE>
```

## Параметры

<MODE> – допустимый режим работы USB-модема [2g, 3g, 4g].

## Значение по умолчанию

Разрешены режимы, которые отображаются как "Allowed modes:" в выводе команды `show cellular status modem <ID>`.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-CELLULAR-MODEM

## Пример

```
esr(config-cellular-modem)# allowed-mode 4g
```

## apn

Данной командой задается точка доступа мобильной сети.

## Синтаксис

```
apn <NAME>
```

## Параметры

<NAME> – точка доступа мобильной сети, задается строкой до 253 символов.

## Значение по умолчанию

Не сконфигурирована.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-CELLULAR-PROFILE

## Пример

```
esr(config-cellular-profile)# apn internet
```

## cellular modem

Данной командой создается USB-модем с определенным идентификатором и осуществляется переход в режим конфигурирования USB-модема.

Использование отрицательной формы команды (no) удаляет сконфигурированный USB-модем.

### Синтаксис

```
[no] cellular modem <ID>
```

### Параметры

<ID> – идентификатор USB-модема в системе [1..10].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# cellular modem 1
```

## cellular profile

Данной командой создается профиль настроек для USB-модема с определенным идентификатором и осуществляется переход в режим конфигурирования профиля.

Использование отрицательной формы команды (no) удаляет сконфигурированный профиль настроек USB-модема.

### Синтаксис

```
[no] cellular profile <ID>
```

### Параметры

<ID> – идентификатор профиля настроек для USB-модема в системе [1..10].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

## Пример

```
esr(config)# cellular profile 1
```

## device

Данной командой задается идентификатор USB-порта подключенного модема.

Использование отрицательной формы команды (no) удаляет установленный идентификатор.

## Синтаксис

```
device <WORD>
```

```
no device
```

## Параметры

<WORD> – идентификатор USB-порта подключенного модема, задается строкой длиной от 1 до 12 символов.

Данный идентификатор должен совпадать с идентификатором в поле "USB port device" в выводе команды show cellular status modem.

## Значение по умолчанию

Не указан.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-CELLULAR-MODEM

## Пример

```
esr(config-cellular-modem)# device 1-1
```

## enable

Данной командой активируется USB-модем.

Использование отрицательной формы команды (no) деактивирует USB-модем.

## Синтаксис

```
[no] enable
```



## Параметры

Команда не содержит параметров.

## Значение по умолчанию

USB-модем не активирован.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-CELLULAR-MODEM

## Пример

```
esr(config-cellular-modem)#
```

## ip-version

Данной командой ограничивается возможность использования семейств IP-адресов в мобильной сети.

Использование отрицательной формы команды (no) разрешает использование адресаций обоих семейств IPv4 и IPv6.

## Синтаксис

```
ip-version { ipv4 | ipv6 }
```

```
no ip-version
```

## Параметры

ipv4 – семейство IPv4;

ipv6 – семейство IPv6;

## Значение по умолчанию

Разрешена адресация обоих семейств IPv4 и IPv6.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-CELLULAR-PROFILE

## Пример

```
esr(config-cellular-profile)# ip-version ipv4
```

## mode

Данной командой задается режим работы беспроводного модема.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
mode <MODE>
```

```
no mode
```

## Параметры

<MODE> – режим работы модема, принимает значения stick/hilink.

## Значение по умолчанию

stick

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-CELLULAR-MODEM

## Пример

```
esr(config-cellular-modem)# mode hilink
```

## mru

Данной командой задается размер максимального принимаемого пакета.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

## Синтаксис

```
mru { <MRU> }
```

```
no mru
```

## Параметры

<MRU> – значение MRU, принимает значения в диапазоне [128..16383].

**Значение по умолчанию**

1500

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-CELLULAR-MODEM

**Пример**

```
esr(config-cellular-modem)# mru 1476
```

**number**

Данной командой устанавливается номер дозвона для подключения к мобильной сети.

Использование отрицательной формы команды (no) удаляет номер дозвона для подключения к мобильной сети.

**Синтаксис**

```
number <WORD>
```

```
no number
```

**Параметры**

<WORD> – номер дозвона для подключения к мобильной сети, задаётся строкой от 1 до 15 символов.

**Значение по умолчанию**

Номер не задан.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-CELLULAR-PROFILE

**Пример**

```
esr(config-cellular-profile)# number *99#
```

## password

Команда для установки пароля пользователя мобильной сети в открытой или зашифрованной форме. Пароль пользователя хранится в конфигурации в зашифрованной форме. При конфигурировании можно задать пароль в открытой форме либо скопировать пароль в зашифрованной форме с другого устройства.

Использование отрицательной формы команды (no) удаляет пароль пользователя.


### Синтаксис

```
password ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no password
```

### Параметры

<CLEAR-TEXT> – пароль в открытой форме, задаётся строкой [1..64] символов, может включать символы [0-9a-fA-F];

<ENCRYPTED-TEXT> – пароль в зашифрованной форме, задаётся строкой [2..128] символов.

 Пароли хранятся в конфигурации в зашифрованной форме независимо от формата, использованного при вводе команды.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-PPP-USER

### Пример

```
esr(config-ppp-user)# password ascii-text 01234567
```

## pin

Данной командой задается код разблокировки SIM-карты.

Использование отрицательной формы команды (no) удаляет код разблокировки SIM-карты.

### Синтаксис

```
pin <WORD>
no pin
```

### Параметры

<WORD> – код разблокировки SIM-карты [4..8] символов. Возможно использование только цифр.

**Значение по умолчанию**

PIN не задан.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-CELLULAR-MODEM

**Пример**

```
esr(config-cellular-modem)# pin 4856
```

**preferred-mode**

Данной командой задается предпочтительный режим работы USB-модема в мобильной сети.

Использование отрицательной формы команды (no) удаляет предпочтительный режим работы USB-модема в мобильной сети.

**Синтаксис**

```
preferred-mode { <MODE> }  
no preferred-mode
```

**Параметры**

<MODE> – предпочтительный режим работы USB-модема [2g, 3g, 4g].

**Значение по умолчанию**

Определяется USB-модемом.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-CELLULAR-MODEM

**Пример**

```
esr(config-cellular-modem)# preferred-mode 4g
```

**profile**

Данной командой на USB-модем назначается созданный профиль настроек для USB-модема.

Использование отрицательной формы команды (no) удаляет назначенный профиль настроек для USB-модема.

### Синтаксис

```
profile <ID>
```

```
no profile
```

### Параметры

<ID> – идентификатор профиля настроек для USB-модема в системе [1..10].

### Значение по умолчанию

ID профиля настроек не задан.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-CELLULAR-MODEM

### Пример

```
esr(config-cellular-modem)# profile 1
```

### show cellular configuration modem

Данной командой отображается информация о сконфигурированных USB-модемах.

### Синтаксис

```
show cellular configuration modem [ <ID> ]
```

### Параметры

<ID> – идентификатор USB-модема в системе [1..10].

Без указания номера USB-модема выводится таблица с краткой информацией о всех сконфигурированных USB-модемах.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

## Пример

```

esr# show cellular configuration modem
Number      State      Description                               USB port device  Profile
-----
1           Enabled   megafon                                  1-1              1
2           Enabled   mts                                       1-2              2

esr# show cellular configuration modem 2
State:      Enabled
Description: mts
USB port device: 1-2
Pin:        --
MRU:        --
MTU:        1500
Preferred mode: none
Allowed modes: all
Profile:    2
Description: MTS
  User name: mts
  Number:    *99#
  APN:       internet.mts.ru
  Password (encrypted): 91A010
  IP version: both
  Allowed auth: EAP
  Security zone: --

```

## show cellular configuration profile

Данной командой отображается информация о сконфигурированных профилях USB-модемов.

### Синтаксис

```
show cellular configuration profile [ <ID> ]
```

### Параметры

<ID> – идентификатор USB-модема в системе [1..10].

Без указания номера USB-модема выводится таблица с краткой информацией о всех сконфигурированных профилях USB-модемов.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

## Пример

```

esr# show cellular configuration profile
Number  User name  APN  Number  Description

```

```

profile
-----
-----
1      gdata      internet      *99#          MEGAFON
2      mts        internet.mts.ru *99#          MTS

esr# show cellular configuration profile 1
Description:      MEGAFON
User name:        gdata
Number:           *99#
APN:              internet
Password (encrypted): 9BB00279B1
IP version:       both
Allowed auth:     EAP

```

## show cellular status modem

Данной командой отображается информация о статусе подключенных USB-модемов.

### Синтаксис

```
show cellular status modem [ <ID> ]
```

### Параметры

<ID> – идентификатор USB-модема в системе [1..10].

Без указания номера USB-модема выводится таблица с краткой информацией о всех обнаруженных USB-модемах.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```

esr# show cellular status modem
Number  USB port  Manufacturer  Model      Current state  Inteface      Link
device  -----  -----      -----      -----      -----      state
-----  -----  -----      -----      -----      -----      -----
1        1-1       huawei        E3372      disabled      --            Down

esr# show cellular status modem 1
Interface 'modem 1' status information:
  USB port device:      1-1
  Manufacturer:        huawei
  Model:                E3372
  Revision:             21.180.01.00.00
  IMEI:                 861821036192893
  Status SIM lock:     --
  Status unlock retries: sim-pin (3) sim-pin2 (3) sim-puk (10) sim-puk2 (10)
  Current state:       disabled
  Access tech:         unknown

```



```

Signal level:          0
Support modes:
  allowed 2G; preferred none;
  allowed 3G; preferred none;
  allowed 4G; preferred none;
  allowed 2G 3G 4G; preferred none;
Allowed modes:         2G 3G 4G
Preferred modes:       none
Type IP:               IPv4
Operator name:         unknown
Registration:          unknown

```

**user**

Данной командой задается имя пользователя мобильной сети. После выполнения данной команды маршрутизатор переходит в режим конфигурирования параметров пользователя.

Использование отрицательной формы команды (no) удаляет имя пользователя мобильной сети.

**Синтаксис**

```
[no] user <NAME>
```

**Параметры**

<NAME> – имя пользователя, задается строкой от 1 до 31 символа.

**Значение по умолчанию**

Пользователь не создан.

**Необходимый уровень привилегий**

15


**Командный режим**

CONFIG-CELLULAR-PROFILE

**Пример**

```
esr(config-cellular-profile)# user gdata
```

**Настройка телефонных портов**

 Данная функция доступна только на маршрутизаторах ESR-12V/12VF/15VF.

**authentication name**

Данная команда позволяет назначить имя пользователя, используемое для аутентификации на SIP-сервере (и сервере регистрации).

Использование отрицательной формы команды (no) удаляет значение.

**Синтаксис**

```
authentication name {<LOGIN> | as-phone}
```

```
no authentication name
```

**Параметры**

<LOGIN> – имя пользователя, используемое для аутентификации, задаётся строкой до 31 символа или директивой as-phone, которая означает, что имя пользователя для аутентификации будет равно телефонному номеру;

As-phone – имя пользователя для аутентификации, равно телефонному номеру.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

15

**Командный режим**

```
CONFIG-VOICE-PORT-FXO
```

```
CONFIG-VOICE-PORT-FXS
```

**Пример**

```
esr(config-voice-port-fxo)# authentication name userlogin
```

**authentication password**

Данная команда позволяет назначить пароль, используемый для аутентификации на SIP-сервере (и сервере регистрации).

Использование отрицательной формы команды (no) удаляет значение.

**Синтаксис**

```
authentication password { <PASS> | encrypted <ENCRYPTED-PASS> }
```

```
no authentication password
```

**Параметры**

<PASS> – пароль для аутентификации, задаётся строкой до 16 символов;

<ENCRYPTED-PASS> – хеш пароля по алгоритму sha512, задаётся строкой размером [2..32] символов.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-VOICE-PORT-FXO

CONFIG-VOICE-PORT-FXS

**Пример**

```
esr(config-voice-port-fxs)# authentication password superpassword
```

**call-forwarding busy**

Данная команда позволяет разрешить использование услуги CFB (Call Forward at Busy) – переадресация вызова при занятости абонента на указанный номер.

Использование отрицательной формы команды (no) запрещает использование услуги «*Переадресация вызова при занятости*».

**Синтаксис**

```
[no] call-forwarding busy
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Услуга CFB отключена.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-VOICE-PORT-FXS

**Пример**

```
esr(config-voice-port-fxs)# call-forwarding busy
```

**call-forwarding busy number**

Данная команда позволяет назначить номер, на который перенаправляются входящие вызовы при занятости абонента, при включенной услуге «*Переадресация вызова при занятости*».

Использование отрицательной формы команды (no) удаляет номер переадресации при занятости.

**Синтаксис**

```
call-forwarding busy number <PHONE>
no call-forwarding busy number
```

**Параметры**

<PHONE> – номер, на который перенаправляются входящие вызовы при занятости абонента, задаётся строкой до 50 символов.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-VOICE-PORT-FXS

**Пример**

```
esr(config-voice-port-fxs)# call-forwarding busy number 4596
```

**call-forwarding no-answer**

Данная команда позволяет разрешить использование услуги *CFNA (Call Forward at No Answer)* – переадресация вызова при неответе абонента на указанный номер.

Использование отрицательной формы команды (no) запрещает использование услуги «*Переадресация вызова при неответе*».

**Синтаксис**

```
[no] call-forwarding no-answer
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Услуга CFNA отключена.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-VOICE-PORT-FXS

## Пример

```
esr(config-voice-port-fxs)# call-forwarding no-answer
```

### call-forwarding no-answer number

Данная команда позволяет назначить номер, на который перенаправляются входящие вызовы при неответе абонента, при включенной услуге «*Переадресация вызова при неответе*».

Использование отрицательной формы команды (no) удаляет номер переадресации при неответе.

### Синтаксис

```
call-forwarding no-answer number <PHONE>
```

```
no call-forwarding no-answer number
```

### Параметры

<PHONE> – номер, на который перенаправляются входящие вызовы при неответе абонента, задаётся строкой до 50 символов.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-VOICE-PORT-FXS

## Пример

```
esr(config-voice-port-fxs)# call-forwarding no-answer number 4685
```

### call-forwarding no-answer timeout

Данная команда позволяет указать интервал времени, через который будет производиться переадресация вызова в случае неответа абонента, при включенной услуге «*Переадресация вызова при неответе*».

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
call-forwarding no-answer timeout <TIME>
```

```
no call-forwarding no-answer timeout
```

## Параметры

<TIME> – интервал времени в секундах, через который будет производиться переадресация вызова в случае неответа абонента, может принимать значения [0-120].

## Значение по умолчанию

0

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-VOICE-PORT-FXS

## Пример

```
esr(config-voice-port-fxs)# call-forwarding no-answer timeout 15
```

## call-forwarding unconditional

Данная команда позволяет разрешить использование услуги *CFU (Call Forward Unconditional)* – все входящие вызовы перенаправляются на указанный номер безусловной переадресации.

Использование отрицательной формы команды (no) запрещает использование услуги «*Переадресация вызова при занятости*».

## Синтаксис

```
[no] call-forwarding unconditional
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Услуга CFU отключена.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-VOICE-PORT-FXS

## Пример

```
esr(config-voice-port-fxs)# call-forwarding unconditional
```

### call-forwarding unconditional number

Данная команда позволяет назначить номер безусловной переадресации.

Использование отрицательной формы команды (no) удаляет номер безусловной переадресации.

#### Синтаксис

```
call-forwarding unconditional number <PHONE>
```

```
no call-forwarding unconditional number
```

#### Параметры

<PHONE> – номер, на который перенаправляются все входящие вызовы, при включенной услуге «*Безусловная переадресация*», задаётся строкой до 50 символов.

#### Значение по умолчанию

Отсутствует.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-VOICE-PORT-FXS

#### Пример

```
esr(config-voice-port-fxs)# call-forwarding unconditional number 4685
```

### call-waiting

Данная команда позволяет разрешить использование услуги «*Ожидание вызова*».

Использование отрицательной формы команды (no) запрещает использование услуги «*Ожидание вызова*».

#### Синтаксис

```
[no] call-waiting
```

#### Параметры

Команда не содержит параметров.

#### Значение по умолчанию

Услуга CW отключена.

#### Необходимый уровень привилегий

10

## Командный режим

CONFIG-VOICE-PORT-FXS

CONFIG-VOIP-PROFILE

## Пример

```
esr(config-voice-port-fxs)# call-waiting
```

## caller-id mode

Данная команда позволяет выбрать режим выдачи номера вызывающего (Caller ID).

Использование отрицательной формы команды (no) отключает выдачу номера вызывающего (Caller ID).

## Синтаксис

```
caller-id mode <MODE>
```

```
no caller-id mode
```

## Параметры

<MODE> – режим выдачи номера вызывающего (Caller ID), может принимать значения:

- auto – автоматическое определение, используемого метода Caller-id. Доступно только на FXO-порту;
- dtmf – определение номера вызывающего абонента методом DTMF. Выдача номера в линию осуществляется между первым и вторым сигналом посылки вызова двухчастотными DTMF-посылками;
- fsk-bell – определение номера и имени вызывающего абонента методом FSK по стандарту Bell202. Выдача номера в линию осуществляется между первым и вторым сигналом посылки вызова потоком данных с частотной модуляцией;
- fsk-v23 – определение номера и имени вызывающего абонента методом FSK по стандарту ITU-T V.23. Выдача номера в линию осуществляется между первым и вторым сигналом посылки вызова потоком данных с частотной модуляцией.

## Значение по умолчанию

Выключает Caller ID.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-VOICE-PORT-FXO

CONFIG-VOICE-PORT-FXS

CONFIG-VOIP-PROFILE



## Пример

```
esr(config-voice-port-fxs)# caller-id mode fsk-v23
```

## dial-type

Данная команда позволяет выбрать режим набора номера FXO-комплектom.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

## Синтаксис

```
dial-type <MODE>
```

```
no dial-type
```

## Параметры

<MODE> – режим набора номера FXO-комплектom, может принимать значения:

- dtmf – набор номера методом передачи тоновых сигналов;
- pulse – набор номера методом последовательного замыкания и размыкания телефонной линии.

## Значение по умолчанию

dtmf

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-VOICE-PORT-FXO

## Пример

```
esr(config-voice-port-fxo)# dial-type pulse
```

## enable

Данной командой активируется профиль или набор правил.

Использование отрицательной формы команды (no) деактивирует профиль или набор правил.

## Синтаксис

```
[no] enable
```

## Параметры

Команда не содержит параметров.

**Значение по умолчанию**

Не активирован.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-VOIP-PROFILE

**Пример**

```
esr(config-config-voip-sip-proxy)# enable
```

**flash call-transfer**

Данная команда позволяет выбрать режим использования функции flash (короткий отбой).  
Использование отрицательной формы команды (no) возвращает значение по умолчанию.

**Синтаксис**

```
flash call-transfer <METHOD>
no flash call-transfer
```

**Параметры**

<METHOD> – режим использования функции flash, может принимать значения:

- attended – flash обрабатывается локально устройством (передача вызова осуществляется после установления соединения с третьим абонентом);
- unattended – flash обрабатывается локально устройством (передача вызова осуществляется по окончании набора номера третьего абонента);
- transmit-flash – передача flash в канал (одним из методов, настроенных командой transfer flash в параметрах SIP-профиля);
- local-transfer – передача вызова внутри устройства, без отправки сообщения REFER.

**Значение по умолчанию**

transmit-flash

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-VOICE-PORT-FXS  
CONFIG-VOIP-PROFILE

## Пример

```
esr(config-voice-port-fxs)# flash call-transfer attended
```

## flash call-transfer mode

Данная команда позволяет выбрать режим передачи вызова – настройка доступна только для Attended calltransfer и Local calltransfer и отвечает за режим активации услуги передачи вызова.

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

## Синтаксис

```
flash call-transfer mode <MODE>
```

```
no flash call-transfer mode
```

## Параметры

<MODE> – режим передачи вызова, может принимать значения:

- r4 – передача вызова активируется после нажатия R 4;
- hook – передача вызова активируется после отбоя;
- both – передача вызова активируется по отбою и по нажатию R 4.

## Значение по умолчанию

r4

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-VOICE-PORT-FXS

CONFIG-VOIP-PROFILE

## Пример

```
esr(config-voice-port-fxs)# flash call-transfer mode hook
```

## flash timer

Данная команда позволяет установить минимальное время обнаружения сигнала flash (короткого отбоя).

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

## Синтаксис

```
flash timer <TIME>
```

```
no flash timer
```

### Параметры

<TIME> – минимальное время обнаружения сигнала flash в миллисекундах, может принимать значения [80..1000].

### Значение по умолчанию

100

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-VOICE-PORT-FXS

CONFIG-VOIP-PROFILE

### Пример

```
esr(config-voice-port-fxs)# flash timer 200
```

### hotline

Данная команда используется для активации услуги «Горячая/Теплая линия». Услуга позволяет автоматически установить исходящее соединение без набора номера сразу после подъема трубки – «горячая линия», либо с задержкой «теплая линия».

### Синтаксис

[no] hotline

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Услуга «Горячая/Теплая линия» отключена.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-VOICE-PORT-FXS

### Пример

```
esr(config-voice-port-fxs)# hotline
```

## hotline ipt

Данная команда используется для активации услуги «Горячая/Теплая линия» в направлении из аналоговой телефонной линии в VoIP.

Использование отрицательной формы команды (no) запрещает использование услуги «Горячая/Теплая линия» в направлении из аналоговой телефонной линии в VoIP.

### Синтаксис

```
[no] hotline ipt
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Услуга «Горячая/Теплая линия» отключена.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-VOICE-PORT-FXO

### Пример

```
esr(config-voice-port-fxo)# hotline ipt
```

## hotline pstn

Данная команда используется для активации услуги «Горячая/Теплая линия» в направлении из VoIP в сторону ТфОП.

Использование отрицательной формы команды (no) запрещает использование услуги «Горячая/Теплая линия» в направлении из VoIP в сторону ТфОП.

### Синтаксис

```
[no] hotline pstn
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Услуга «Горячая/Теплая линия» отключена.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-VOICE-PORT-FXO

## Пример

```
esr(config-voice-port-fxo)# hotline pstn
```

## hotline number

Данная команда позволяет назначить номер телефона, с которым будет устанавливаться соединение при использовании услуги «Горячая/Теплая линия».

Использование отрицательной формы команды (no) удаляет номер услуги «Горячая/Теплая линия».

## Синтаксис

```
hotline number <PHONE>
```

```
no hotline number
```

## Параметры

<PHONE> – номер телефона, с которым будет устанавливаться соединение после поднятия трубки телефона, задаётся строкой от 1 до 50 символов.

## Значение по умолчанию

Значение не задано.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-VOICE-PORT-FXS

## Пример

```
esr(config-voice-port-fxs)# hotline number 5462
```

## hotline number ipt

Данная команда позволяет назначить номер телефона, с которым будет устанавливаться соединение при использовании услуги «Горячая/Теплая линия» в направлении из аналоговой телефонной линии в VoIP.

Использование отрицательной формы команды (no) удаляет номер услуги «Горячая/Теплая линия» в направлении из аналоговой телефонной линии в VoIP.

### Синтаксис

```
hotline number ipt <PHONE>  
no hotline number ipt
```

### Параметры

<PHONE> – номер телефона, на который осуществляется вызов при использовании услуги «Горячая/Теплая линия» в направлении из аналоговой телефонной линии в VoIP, задаётся строкой от 1 до 50 символов.

### Значение по умолчанию

Значение не задано.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-VOICE-PORT-FXO

### Пример

```
esr(config-voice-port-fxo)# hotline number ipt 6347
```

### hotline number pstn

Данная команда позволяет назначить номер телефона, с которым будет устанавливаться соединение при использовании услуги «Горячая/Теплая линия» в направлении из VoIP в аналоговую телефонную линию.

Использование отрицательной формы команды (no) удаляет номер услуги «Горячая/Теплая линия» в направлении из VoIP в аналоговую телефонную линию.

### Синтаксис

```
hotline number pstn <PHONE>  
no hotline number pstn
```

### Параметры

<PHONE> – горячий номер при звонке в сторону ТфОП, задаётся строкой до 50 символов.

### Значение по умолчанию

Значение не задано.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-VOICE-PORT-FXO

**Пример**

```
esr(config-voice-port-fxo)# hotline number pstn 5462
```

**hotline timeout**

Данная команда позволяет установить «Таймаут задержки» перед установлением соединения и после поднятия трубки телефона.

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

**Синтаксис**

```
hotline timeout <TIME>
```

```
no hotline timeout
```

**Параметры**

<TIME> – интервал времени в секундах, через который будет устанавливаться соединение с встречным абонентом, может принимать значения [0..60].

**Значение по умолчанию**

0

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-VOICE-PORT-FXS

**Пример**

```
esr(config-voice-port-fxs)# hotline timeout 10
```

**hotline timeout ipt**

Данная команда позволяет установить «Таймаут задержки» перед установлением соединения с «Горячим номером» в направлении из аналоговой телефонной линии в VoIP.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.



**Синтаксис**

```
hotline timeout ipt <TIME>  
no hotline timeout ipt
```

**Параметры**

<TIME> – интервал времени в секундах, через который будет устанавливаться соединение с встречным абонентом, может принимать значения [0..60].

**Значение по умолчанию**

0

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-VOICE-PORT-FXO

**Пример**

```
esr(config-voice-port-fxo)# hotline timeout ipt 10
```

**hotline timeout pstn**

Данная команда позволяет установить «Таймаут задержки» перед установлением соединения с «Теплым номером» в направлении из VoIP в аналоговую телефонную линию.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
hotline timeout pstn <TIME>  
no hotline timeout pstn
```

**Параметры**

<TIME> – интервал времени в секундах, через который будет устанавливаться соединение с встречным абонентом, может принимать значения [0..60].

**Значение по умолчанию**

0

**Необходимый уровень привилегий**

10

## Командный режим

CONFIG-VOICE-PORT-FXO

### Пример

```
esr(config-voice-port-fxo)# hotline timeout pstn 10
```

### hybrid rx

Данная команда позволяет увеличить усиление сигнала на приеме.

### Синтаксис

```
hybrid rx <NUM>
```

```
no hybrid rx
```

### Параметры

<NUM> – принимает значение от -200 до 200.

### Значение по умолчанию

0

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-VOICE-PORT-FXO

CONFIG-VOICE-PORT-FXS

### Пример

```
esr(config-voice-port-fxs)# hybrid rx 100
```

### hybrid tx

Данная команда позволяет увеличить усиление сигнала на передаче.

### Синтаксис

```
hybrid rx <NUM>
```

```
no hybrid rx
```

### Параметры

<NUM> – принимает значение от -200 до 200.

**Значение по умолчанию**

0

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-VOICE-PORT-FXO

CONFIG-VOICE-PORT-FXS

**Пример**

```
esr(config-voice-port-fxs)# hybrid rx 20
```

**ipt prefix-name**

Данная команда позволяет указать префикс к имени в CallerID, которое передается в направлении VoIP. Использование отрицательной формы команды (no) удаляет установленное значение.

**Синтаксис**

```
ipt prefix-name <NAME>
```

```
no ipt prefix-name
```

**Параметры**

<NAME> – префикс, добавляемый к имени в CallerID, задаётся строкой от 1 до 21 символа.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-VOICE-PORT-FXO

**Пример**

```
esr(config-voice-port-fxo)# ipt prefix-name localPSTN
```

**ipt prefix-number**

Данная команда позволяет указать префикс к номеру в CallerID, который передается в направлении VoIP.

Использование отрицательной формы команды (no) удаляет установленное значение.

**Синтаксис**

```
ipt prefix-number <NUMBER>
no ipt prefix-number
```

**Параметры**

<NUMBER> – префикс, добавляемый к номеру в CallerID, задаётся строкой от 1 до 21 символа.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-VOICE-PORT-FXO

**Пример**

```
esr(config-voice-port-fxo)# ipt prefix-number 7
```

**ipt offhook-ringing**

Данная команда используется для замыкания шлейфа при вызове из TDM в IP до проключения голосового канала в сторону VoIP.

Использование отрицательной формы команды (no) позволяет не замыкать шлейф при вызове из TDM в IP до тех пор, пока не будет получен голосовой тракт до взаимодействующего по протоколу SIP-шлюза. Данная настройка используется только совместно с настройкой «Hotline».

**Синтаксис**

```
[no] ipt offhook-ringing
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Позволяет не замыкать шлейф при вызове из TDM в IP до проключения голосового канала.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-VOICE-PORT-FXO

**Пример**

```
esr(config-voice-port-fxo)# no ipt offhook-ringing
```

**ipt ring-number**

Данная команда позволяет указать количество «Посылок вызова», по которому комплект FXO выполнит замыкание шлейфа («снимет трубку») и выдаст в телефонную линию сигнал «Ответ станции».

Использование отрицательной формы команды (no) возвращается в значение по умолчанию.

**Синтаксис**

```
ipt ring-number <COUNT>
```

```
no ipt ring-number
```

**Параметры**

<COUNT> – количество «Посылок вызова», по которому комплект FXO выполнит замыкание шлейфа, может принимать значения [2..10].

**Значение по умолчанию**

2

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-VOICE-PORT-FXO

**Пример**

```
esr(config-voice-port-fxo)# ipt ring-number 4
```

**profile sip**

Данная команда позволяет выбрать SIP-профиль для конфигурируемого порта.

**Синтаксис**

```
profile sip <PROFILE>
```

**Параметры**

<PROFILE> – индекс SIP-профиля, может принимать значения [1..5].

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-VOICE-PORT-FXO

CONFIG-VOICE-PORT-FXS

**Пример**

```
esr(config-voice-port-fxo)# profile sip 1
```

**profile pbx**

Данная команда позволяет выбрать SIP-профиль для конфигурируемого порта.

**Синтаксис**

```
profile pbx<PROFILE>
```

**Параметры**

<PROFILE> – имя SIP-профиля, созданного на сервере PBX, задаётся строкой от 1 до 31 символа.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-VOICE-PORT-FXO

CONFIG-VOICE-PORT-FXS

## Пример

```
esr(config-voice-port-fxo)# profile pbx fxs_ports
```

## pstn transmit-number

Данная команда используется для указания передачи номера, принятого из IP (из заголовка Request URI запроса INVITE) в линию, за исключением номер порта.

Использование отрицательной формы команд запрещает передачу номера, принятого из IP.

## Синтаксис

```
[no] pstn transmit-number
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Не передавать полный номер.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-VOICE-PORT-FXO

## Пример

```
esr(config-voice-port-fxo)# pstn transmit-number
```

## pstn transmit-prefix

Данная команда используется для указания передачи абонентского номера FXO-комплекта.

Использование отрицательной формы команды запрещает передачу абонентского номера FXO-комплекта.

## Синтаксис

```
[no] pstn transmit-prefix
```

## Параметры

Команда не содержит параметров.

**Значение по умолчанию**

Передавать абонентский номер FXO-комплекта.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-VOICE-PORT-FXO

**Пример**

```
esr(config-voice-port-fxo)# pstn transmit-prefix
```

**shutdown**

Данная команда используется для отключения порта.

Использование отрицательной формы команды (no) включает порт.

**Синтаксис**

[no] shutdown

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Порт включен.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-VOICE-PORT-FXO

CONFIG-VOICE-PORT-FXS

**Пример**

```
esr(config-voice-port-fxs)# shutdown
```

**sip port**

Данная команда используется для указания номера UDP-порта для приёма входящих сообщений SIP на данный аккаунт, а также для отправки исходящих SIP-сообщений с данного аккаунта.



Использование отрицательной формы команды (no) возвращает значение по умолчанию.

### Синтаксис

```
sip port <PORT>
```

```
no sip port
```

### Параметры

<PORT> – номер UDP-порта, принимает значения [1..65535].

### Значение по умолчанию

5060

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-VOICE-PORT-FXO

CONFIG-VOICE-PORT-FXS

### Пример

```
esr(config-voice-port-fxs)# sip port 5080
```

### sip user display-name

Данная команда позволяет назначить имя пользователя, сопоставленное с портом (отображается в поле Display-Name заголовка From в исходящих сообщениях SIP).

Использование отрицательной формы команды (no) удаляет значение.

### Синтаксис

```
sip user display-name <LOGIN>
```

```
no sip user display-name
```

### Параметры

<LOGIN> – имя пользователя, которое отображается в поле Display-Name, задаётся строкой до 31 символа.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-VOICE-PORT-FXO

CONFIG-VOICE-PORT-FXS

## Пример

```
esr(config-voice-port-fxs)# sip user display-name "port-1"
```

## sip user phone

Данная команда позволяет назначить абонентский номер, закрепленный за телефонным портом. Использование отрицательной формы команды (no) удаляет абонентский номер.

## Синтаксис

```
sip user phone <PHONE>
```

```
no sip user phone
```

## Параметры

<PHONE> – абонентский номер, закрепленный за телефонным портом, задается строкой до 50 СИМВОЛОВ.

## Значение по умолчанию

Значение не задано.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-VOICE-PORT-FXO

CONFIG-VOICE-PORT-FXS

## Пример

```
esr(config-voice-port-fxs)# sip user phone 4101
```

## timing delay

Данная команда позволяет установить задержку перед набором номера.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
timing delay <TIME>
```

```
no timing delay
```

### Параметры

<TIME> – задержка перед набором номера в секундах, может принимать значения [0..10].

### Значение по умолчанию

2

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-VOICE-PORT-FXO

### Пример

```
esr(config-voice-port-fxo)# timing delay 4
```

### timing digit

Данная команда позволяет установить минимально детектируемый межцифровой интервал для FXS-порта.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
timing digit <TIME>
```

```
no timing pulse-interdigit
```

### Параметры

<TIME> – минимальный межцифровой интервал в миллисекундах, может принимать значения [150...20000].

### Значение по умолчанию

200

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-VOICE-PORT-FXS

CONFIG-VOIP-PROFILE

## Пример

```
esr(config-voice-port-fxs)# timing interdigit 300
```

### timing flash

Данная команда позволяет установить время замыкания шлейфа для имитации импульса «flash».

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
timing flash <TIME>
```

```
no timing flash
```

### Параметры

<TIME> – время замыкания шлейфа в миллисекундах, может принимать значения [70..1000].

### Значение по умолчанию

100

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-VOICE-PORT-FXO

## Пример

```
esr(config-voice-port-fxo)# timing flash 150
```

### timing pulse-digit

Данная команда позволяет установить длительность импульса при импульсном наборе номера.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
timing pulse-digit <TIME>
```

```
no timing pulse-digit
```

### Параметры

<TIME> – длительность импульса при импульсном наборе номера в миллисекундах, может принимать значения [50..120].

**Значение по умолчанию**

80

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-VOICE-PORT-FXO

**Пример**

```
esr(config-voice-port-fxo)# timing pulse-digit 75
```

**timing pulse-interdigit**

Данная команда позволяет установить межцифровой интервал при импульсном наборе номера в аналоговую телефонную линию для FXO-порта.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
timing pulse-interdigit <TIME>
```

```
no timing pulse-interdigit
```

**Параметры**

<TIME> – межцифровой интервал при импульсном наборе номера в миллисекундах, может принимать значения [80...2500].

**Значение по умолчанию**

200

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-VOICE-PORT-FXO

**Пример**

```
esr(config-voice-port-fxo)# timing pulse-interdigit 300
```

### timing pulse-pause

Данная команда позволяет установить длительность паузы при импульсном наборе номера. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### Синтаксис

```
timing pulse-pause <TIME>  
no timing pulse-pause
```

#### Параметры

<TIME> – длительность паузы при импульсном наборе номера в миллисекундах, может принимать значения [50..100].

#### Значение по умолчанию

80

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-VOICE-PORT-FXO

#### Пример

```
esr(config-voice-port-fxo)# timing pulse-pause 75
```

### timing tone-digit

Данная команда позволяет установить длительность тона при тоновом наборе номера. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### Синтаксис

```
timing tone-digit <TIME>  
no timing tone-digit
```

#### Параметры

<TIME> – длительность тона при тоновом наборе номера в миллисекундах, может принимать значения [65..100].

#### Значение по умолчанию

80

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-VOICE-PORT-FXO

**Пример**

```
esr(config-voice-port-fxo)# timing tone-digit 75
```

**timing tone-interdigit**

Данная команда позволяет установить длительность паузы при тоновом наборе номера. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
timing tone-interdigit <TIME>
```

```
no timing tone-interdigit
```

**Параметры**

<TIME> – длительность паузы при тоновом наборе номера в миллисекундах, может принимать значения [80..2500].

**Значение по умолчанию**

100

**Необходимый уровень привилегий**

10


**Командный режим**

CONFIG-VOICE-PORT-FXO

**Пример**

```
esr(config-voice-port-fxo)# timing tone-interdigit 150
```

**Тестирование телефонных портов**

 Данная функция доступна только на маршрутизаторах ESR-12V/12VF/15VF.

### test voice-port start

Данная команда используется для запуска процедуры измерения электрических характеристик абонентской линии.

#### Синтаксис

```
test voice-port <NUM> start
```

#### Параметры

<NUM> – номер FXS-порта, может принимать значения [1..3].

#### Необходимый уровень привилегий

1

#### Командный режим

ROOT

#### Пример

```
esr# test voice-port 3 start
Voice-port 3 test started. The result will be available in 80 seconds, or more if the test run
on other ports.
```

### test voice-port status

Данная команда используется для вывода результатов крайней процедуры измерения электрических характеристик абонентской линии.

#### Синтаксис

```
test voice-port <NUM> status
```

#### Параметры

<NUM> – номер FXS-порта, может принимать значения [1..3].

#### Необходимый уровень привилегий

1

#### Командный режим

ROOT

#### Пример

```
esr# test voice-port 3 status
Testing voice-port 3 is idle
```



Last test start: Sat May 20 16:01:37 2017

Number voip-port 3

Foreign DC voltage A (TIP):	0.128377 U
Foreign DC voltage B (RING):	0.144342 U
Foreign AC voltage A (TIP):	0.026239 U
Foreign AC voltage B (RING):	0.032287 U
Cross current:	0.260343 mA
Longitudinal current:	-0.101857 mA
Line supply voltage:	-50.370598 U
Resistance A (TIP) - B (RING):	1007.203674 kOm
Resistance A (TIP) - Ground:	402.105469 kOm
Resistance B (RING) - Ground:	874.041443 kOm
Capacity A (TIP) - B (RING):	50.000000 nF
Capacity A (TIP) - Ground:	573.000000 nF

## 15 Управление L2-функциями

- Управление L2-функциями
  - bridge
  - bridge-group
  - description
  - enable
  - mac-address
  - ports vrrp filtering
  - protected-ports
  - protected-ports exclude
  - show interfaces bridge
  - thresholds wifi-tunnels-number high
  - thresholds wifi-tunnels-number low
  - unknown-unicast-forwarding disable
  - vlan
- Управление Spanning Tree
  - instance
  - name
  - revision
  - show spanning-tree
  - spanning-tree
  - spanning-tree bpdu
  - spanning-tree cost
  - spanning-tree disable
  - spanning-tree guard root
  - spanning-tree forward-time
  - spanning-tree hello-time
  - spanning-tree link-type
  - spanning-tree max-age
  - spanning-tree mode
  - spanning-tree mst configuration
  - spanning-tree mst cost
  - spanning-tree mst max-hops
  - spanning-tree mst port-priority
  - spanning-tree pathcost method
  - spanning-tree portfast
  - spanning-tree port-priority
  - spanning-tree priority
- Настройка и мониторинг VLAN
  - force-up
  - ip internal-usage-vlan
  - name
  - show interfaces switch-port vlans
  - show vlans
  - show vlans internal-usage
  - switchport access vlan
  - switchport default-vlan tagged
  - switchport forbidden default-vlan
  - switchport general acceptable-frame-type
  - switchport general allowed vlan
  - switchport general allowed vlan auto-all
  - switchport general ingress-filtering disable
  - switchport general macs-group
  - switchport general pvid
  - switchport macs-group
  - switchport trunk allowed vlan

- [switchport trunk allowed vlan auto-all](#)
- [switchport trunk native-vlan](#)
- [vlan](#)
- [Настройка и мониторинг Voice-VLAN](#)
  - [application](#)
  - [dscp](#)
  - [lldp network-policy](#)
  - [network-policy](#)
  - [priority](#)
  - [vlan](#)

## Управление L2-функциями

### bridge

Данной командой добавляется сетевой мост в систему и осуществляется переход в режим настройки его параметров.

Использование отрицательной формы команды (no) удаляет мост.

#### Синтаксис

```
[no] bridge <BRIDGE-ID>
```

#### Параметры

<BRIDGE-ID> – идентификационный номер моста, задается в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG

#### Пример

Переход в режим конфигурирования сетевого моста *bridge 10*:

```
esr(config)# bridge 10
esr(config-bridge)#
```

### bridge-group

Данная команда используется для добавления в текущего сетевого интерфейса в L2-домен.

Использование отрицательной формы команды (no) удаляет интерфейс из L2-домена.

#### Синтаксис

```
bridge-group <BRIDGE-ID> [tagged]
no bridge-group
```

## Параметры

<BRIDGE-ID> – идентификационный номер моста, задается в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

tagged – ключ для направления тегированных кадров физического интерфейса в bridge. Применимо только на физических интерфейсах маршрутизаторов ESR-10/12V/12VF/15/15R/15VF/20/21/30/31/100/200/3100/3200/3200L/3300.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-L2TPV3

CONFIG-GRE

CONFIG-SUBTUNNEL

CONFIG-OPENVPN

## Пример

```
esr(CONFIG-IF-SUB)# bridge-group 15
```

## description

Данная команда используется для назначения описания конфигурируемому сетевому мосту.

Использование отрицательной формы команды (no) удаляет установленное описание.

## Синтаксис

```
description <DESCRIPTION>
```

```
no description
```

## Параметры

<DESCRIPTION> – описание сетевого моста, задается строкой до 255 символов.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-BRIDGE

## Пример

```
esr(config-bridge)# description "broadway"
```

## enable

Данной командой разрешается работа сетевого моста. Без указания данной команды сетевой мост не работает.

Использование отрицательной формы команды (no) отключает маршрутизацию данных.

## Синтаксис

[no] enable

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Выключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-BRIDGE

## Пример

```
esr(config-bridge)# enable
```

## mac-address

Данная команда позволяет задать MAC-адрес сетевого моста, физического или агрегированного интерфейса, отличного от системного.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
mac-address <ADDR>
```

```
no mac-address
```

**Параметры**

<ADDR> – MAC-адрес сетевого моста, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF].

**Значение по умолчанию**

Системный MAC-адрес.

**Необходимый уровень привилегий**

15

**Командный режим**

```
CONFIG-IF-GI
```

```
CONFIG-IF-TE
```

```
CONFIG-IF-TWE
```

```
CONFIG-IF-FO
```

```
CONFIG-IF-HU
```

```
CONFIG-IF-OOB
```

```
CONFIG-IF-PORT-CHANNEL
```

```
CONFIG-BRIDGE
```

**Пример**

```
esr(config-bridge)# mac-address A8:F9:B0:00:00:04
```

**ports vrrp filtering**

Данной командой включается запрет на отправку VRRP-сообщений в интерфейсах, включенных в L2-домен.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
[no] ports vrrp filtering {enable | exclude vlan}
```

**Параметры**

enable – включение запрета на отправку VRRP-сообщений в интерфейсах, включенных в L2-домен;

exclude vlan – при указании данного параметра VLAN исключается из списка фильтруемых.

**Значение по умолчанию**

Фильтрация VRRP-сообщений отключена.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-BRIDGE

**Пример**

```
esr(config-bridge)# ports vrrp filtering enable
```

**protected-ports**

Данной командой осуществляется управление режимом изоляции интерфейсов.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
protected-ports <MODE>
```

**Параметры**

<MODE> – режим изоляции интерфейсов, принимает следующие значения:

- none – изоляция интерфейсов отключена. В данном режиме коммутация кадров между членами сетевого моста разрешена;
- local – изоляция интерфейсов включена. В данном режиме коммутация кадров между членами сетевого моста запрещена;
- radius – изоляция интерфейсов включена. Для использования данного режима требуется настройка Wi-Fi контроллера туннелей в режиме "radius". В данном режиме коммутация кадров между членами сетевого моста запрещена, за исключением SoftGRE DATA-туннелей. Для SoftGRE DATA-туннелей параметры изоляции запрашиваются у RADIUS-сервера.

**Значение по умолчанию**

none

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-BRIDGE

## Пример

```
esr(config-bridge)# protected-ports local
```

## protected-ports exclude

Данной командой исключаются их списка изолируемых сущности, включенные в сетевой мост. Данная опция актуальна при настройке режима **protected-ports** отличного от значения по умолчанию.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
[no] protected-ports exclude { <IF> | <TUN> | <VLAN> }
```

## Параметры

<IF> – интерфейс, задается в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

<TUN> – имя туннеля, задается в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#).

<VLAN> – vlan, связанный с сетевым мостом.

## Значение по умолчанию

Отключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-BRIDGE

## Пример

```
esr(config-bridge)# protected-ports exclude vlan
```

## show interfaces bridge

Данная команда используется для просмотра информации о VLAN, интерфейсах, туннелях, объединенных мостом.

## Синтаксис

```
show interfaces bridge [<BRIDGE-ID>]
```



## Параметры

<BRIDGE-ID> – идентификационный номер моста, задается в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
esr# show interfaces bridge 1
Bridges      Interfaces
-----
bridge 1     vlan 1,gil/0/1.10
```

## thresholds wifi-tunnels-number high

 Данная функция поддерживается только при наличии лицензии на Wi-Fi controller.

Данной командой устанавливается верхний порог аварии превышения числа включенных саб-туннелей softgre в bridge.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

## Синтаксис

```
thresholds wifi-tunnels-number high<NUM>
no thresholds wifi-tunnels-number high
```

## Параметры

<NUM> – верхний порог аварии превышения числа включенных саб-туннелей softgre в bridge, принимает значение в диапазоне [1..1000].

## Значение по умолчанию

950

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-BRIDGE

## Пример

```
esr(config)# thresholds wifi-tunnels-number high 200
```

## thresholds wifi-tunnels-number low

 Данная функция поддерживается только при наличии лицензии на Wi-Fi controller.

Данной командой устанавливается нижний порог аварии превышения числа включенных саб-туннелей softgre в bridge.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

## Синтаксис

```
thresholds wifi-tunnels-number low <NUM>
no thresholds wifi-tunnels-number low
```

## Параметры

<NUM> – нижний порог аварии превышения числа включенных саб-туннелей softgre в bridge, принимает значение в диапазоне [1..1000].

## Значение по умолчанию

925

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-BRIDGE

## Пример

```
esr(config)# thresholds wifi-tunnels-number low 20
```

## unknown-unicast-forwarding disable

Данной командой запрещается коммутация unicast-трафика с отсутствующими MAC-адресами в таблице MAC-адресов сетевого моста.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
[no] unknown-unicast-forwarding disable
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Коммутация unicast-трафика с неизвестными MAC-адресами разрешена.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-BRIDGE

## Пример

```
esr(config-bridge)# unknown-unicast-forwarding disable
```

## vlan

Данная команда используется для связывания текущего сетевого моста с VLAN. Все порты, являющиеся членами назначаемого VLAN, автоматически включаются в сетевой мост и становятся участниками общего L2-домена. Для управления членством сетевых интерфейсов в VLAN используются команды, описанные в разделе [Настройка и мониторинг VLAN](#).

Использование отрицательной формы команды (no) удаляет привязку VLAN и отключает соответствующие интерфейсы от сетевого моста.

## Синтаксис

```
vlan <VID>
```

```
no vlan
```

## Параметры

<VID> – идентификатор VLAN, задаётся в диапазоне [1..4095].

## Необходимый уровень привилегий

10

## Командный режим


CONFIG-BRIDGE

## Пример

```
esr(config-vlan)# vlan 40
```

## Управление Spanning Tree

### instance

 В текущей версии ПО данная команда поддерживается только на маршрутизаторе ESR-1000.

Данной командой создается соответствие между экземпляром протокола MSTP и группами VLAN. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### Синтаксис

```
[no] instance <INSTANCE> vlan <VID>
```

#### Параметры

<INSTANCE> – идентификатор MST-экземпляра, указывается в диапазоне [1..15];

<VID> – идентификационный номер VLAN, задаётся в диапазоне [1..4094].

#### Необходимый уровень привилегий

10


#### Командный режим

CONFIG-MST

#### Пример

```
esr(config-mst)#instance 5 vlan 10-250
```

### name

 В текущей версии ПО данная команда поддерживается только на маршрутизаторе ESR-1000.

Данной командой указывается имя конфигурации MSTP.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### Синтаксис

```
name <NAME>
```

```
no name
```

#### Параметры

<NAME> – имя конфигурации MSTP, задаётся строкой до 31 символа.

#### Необходимый уровень привилегий

10


## Командный режим

CONFIG-MST

### Пример

```
esr(config-mst)# name test
```

### revision

 В текущей версии ПО данная команда поддерживается только на маршрутизаторе ESR-1000.

Данной командой задается номер ревизии конфигурации MSTP.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
revision <NUM>
```

```
no revision
```

### Параметры

<NUM> – номер ревизии конфигурации MSTP, задается в диапазоне [0..65535].

### Значение по умолчанию

0

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-MST

### Пример

```
esr(config-mst)#revision 5000
```

### show spanning-tree

Данная команда отображает подробную информацию о настройке протокола семейства STP.

### Синтаксис

```
show spanning-tree { <IF> | tunnel <TUN> | bpdu | bridge { vrf <VRF_NAME> | global  
[active] | <BRIDGE_NUM> [ active ] [ vrf <VRF_NAME> ] } }
```

## Параметры

<IF> – физический или агрегированный интерфейс или группа интерфейсов, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#);

<BRIDGE> – номер сконфигурированного bridge;

<VRF\_NAME> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать протокол маршрутизации;

global – при указании команды отображается информация о состоянии протокола STP/RSTP/MSTP;

active – при указании команды отображается информация о состоянии протокола STP/RSTP/MSTP, а также дополнительные сведения о состоянии интерфейсов, их типов, ролей, стоимости и приоритета;

bpdu – при указании команды отображается информация о работе устройства с BPDU: filtering – пакеты BPDU будут отброшены, flooding – пакеты BPDU будут рассылаться в широковещательный домен;

bridge – при указании команды отображается информация о состоянии протокола VSTP для соответствующего bridge-домена.

## Необходимый уровень привилегий

1

## Командный режим


ROOT

## Пример

```
esr# show spanning-tree gigabitethernet 1/0/10
Port gi1/0/10 disabled
State: BLK
Port id: ---
Type: ---
Designated bridge Priority: ---
Designated port id: ---
Role: ---
Port cost: ---
Designated path cost: ---
Address: ---
Port Fast: ---
esr# show spanning-tree bridge
Protocol version: STP
  Root ID: [32768] 02:01:02:03:04:55
    Root port: [128] gigabitethernet 1/0/14
    Pathcost 4
    Message Age 1
    Hello time: 2 Max age time: 20 Forward delay: 15
  Bridge ID: [32768] 02:20:03:A0:04:90
    Hello time: 2 Max age time: 20 Forward delay: 15
    Transmit hold count: 6 Topology change: 0
    Time since topology change: 13736 Topology change count: 2 show
```

## spanning-tree

Команда активирует работу протоколов семейства Spanning Tree (STP, RSTP, MSTP).

 Протокол MSTP поддерживается только на маршрутизаторе ESR-1000.

Использование отрицательной формы команды (no) выключает поддержку протоколов семейства Spanning Tree.

### Синтаксис

```
[no] spanning-tree
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Включено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# spanning-tree
```

### spanning-tree

Команда активирует работу протокола VSTP для широковещательного домена, организованного с помощью bridge.

Использование отрицательной формы команды (no) выключает поддержку протокола VSTP для широковещательного домена, организованного с помощью bridge.

### Синтаксис

```
[no] spanning-tree
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Включено.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-BRIDGE

### Пример

```
esr(config-bridge)# spanning-tree
```

### spanning-tree bpdu

Данной командой определяется режим обработки пакетов BPDU интерфейсом, на котором выключен протокол STP.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
spanning-tree bpdu <MODE>
```

```
no spanning-tree bpdu
```

### Параметры

<MODE> – режим работы:

- filtering – на интерфейсе с выключенным протоколом STP BPDU-пакеты фильтруются;
- flooding – на интерфейсе с выключенным протоколом STP нетегированные BPDU-пакеты передаются, тегированные – фильтруются.

### Значение по умолчанию

flooding

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-GRE

CONFIG-SUBTUNNEL

CONFIG-L2TPV3



CONFIG-OPENVPN

CONFIG-OPENVPN-SERVER

**Пример**

```
esr(config-if-gi)# spanning-tree bpdu filtering
```

**spanning-tree cost**

Данной командой устанавливается размер, на который увеличивается поле cost при коммутации DPDU-пакета.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
spanning-tree cost <COST>
```

```
no spanning-tree cost
```

**Параметры**

<COST> – стоимость пути, устанавливается в диапазоне [1..20000000].

**Значение по умолчанию**

4

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-GRE

CONFIG-SUBTUNNEL

CONFIG-L2TPV3

CONFIG-OPENVPN

CONFIG-OPENVPN-SERVER

## Пример

```
esr(config-if-gi)# spanning-tree cost 115
```

## spanning-tree disable

Данной командой запрещается работа протокола STP на конфигурируемом интерфейсе. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
[no] spanning-tree disable
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

STP на интерфейсе включен.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-GRE

CONFIG-SUBTUNNEL

CONFIG-L2TPV3

CONFIG-OPENVPN

CONFIG-OPENVPN-SERVER

## Пример

```
esr(config-if-gi)# spanning-tree disable
```

## spanning-tree guard root

Данной командой включается блокировка получения BPDU с более высоким STP-приоритетом. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
[no] spanning-tree guard root
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-GRE

CONFIG-SUBTUNNEL

CONFIG-L2TPV3

CONFIG-OPENVPN

CONFIG-OPENVPN-SERVER

### Пример

```
esr(config-if-gi)# spanning-tree guard root
```

## spanning-tree forward-time

Данной командой устанавливается интервал времени, затрачиваемый на прослушивание и изучение состояний перед переключением в состояние передачи.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
spanning-tree forward-time <TIME>  
no spanning-tree forward-time
```

**Параметры**

<TIME> – время в секундах, принимает значения [4..30].

**Значение по умолчанию**

15

**Необходимый уровень привилегий**

10

**Командный режим**

```
CONFIG  
CONFIG-BRIDGE
```

**Пример**

```
esr(config)# spanning-tree forward-time 20
```

**spanning-tree hello-time**

Данной командой устанавливается интервал времени между отправкой BPDU-пакетов. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
spanning-tree hello-time <TIME>  
no spanning-tree hello-time
```

**Параметры**

<TIME> – время в секундах, принимает значения [1..10].

**Значение по умолчанию**

2

**Необходимый уровень привилегий**

10

## Командный режим

CONFIG

CONFIG-BRIDGE

## Пример

```
esr(config)# spanning-tree hello-time 20
```

## spanning-tree link-type

Данной командой устанавливается протокол RSTP в передающее состояние и определяет тип связи для выбранного порта – «точка-точка», «разветвлённый».

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
spanning-tree link-type { point-to-point | shared }  
no spanning-tree link-type
```

## Параметры

point-to-point – команда определяет интерфейс как «точка-точка»;

shared – команда определяет интерфейс как «разветвленный».

## Значение по умолчанию

point-to-point

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-GRE

CONFIG-SUBTUNNEL

CONFIG-L2TPV3

CONFIG-OPENVPN

CONFIG-OPENVPN-SERVER

### Пример

```
esr(config-if-gi)# spanning-tree link-type point-to-point
```

### spanning-tree max-age

Данной командой устанавливается время жизни связующего дерева STP.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
spanning-tree max-age <TIME>
```

```
no spanning-tree max-age
```

### Параметры

<TIME> – время в секундах, принимает значения [6..40].

### Значение по умолчанию

20

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

CONFIG-BRIDGE

### Пример

```
esr(config)# spanning-tree max-age 35
```

### spanning-tree mode

Данной командой выбирается поддерживаемый протокол из семейства STP для широковещательного домена, организованного с помощью bridge или физических интерфейсов маршрутизатора в режиме switchport.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
spanning-tree mode <MODE>
```

```
no spanning-tree mode
```

## Параметры

<MODE> – протокол семейства STP:

- STP – IEEE 802.1D Spanning Tree Protocol;
- RSTP – IEEE 802.1W Rapid Spanning Tree Protocol;
- MSTP – IEEE 802.1s Multiple Spanning Trees (доступно только в глобальной конфигурации на ESR-1000).

## Значение по умолчанию

RSTP

## Необходимый уровень привилегий

10

## Командный режим


CONFIG

CONFIG-BRIDGE

## Пример

```
esr(config)# spanning-tree mode STP
```

## spanning-tree mst configuration

 В текущей версии ПО данная команда поддерживается только на маршрутизаторе ESR-1000.

Данной командой осуществляется переход в режим конфигурирования MSTP-параметров.

## Синтаксис

```
spanning-tree mst configuration
```

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# spanning-tree mst configuration
```

```
esr(config-mst)#
```

## spanning-tree mst cost

**!** В текущей версии ПО данная команда поддерживается только на маршрутизаторе ESR-1000.

Данной командой устанавливается метод определения ценности пути для экземпляра MST. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
spanning-tree mst <INSTANCE> cost <COST>  
no spanning-tree mst <INSTANCE>
```

### Параметры

<INSTANCE> – идентификатор MST-экземпляра, указывается в диапазоне [1..15];

<COST> – стоимость пути, устанавливается в диапазоне [1..20000000].

### Значение по умолчанию

4

### Необходимый уровень привилегий

10

### Командный режим


```
CONFIG  
CONFIG-IF-GI  
CONFIG-TE  
CONFIG-IF-PORT-CHANNEL  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-GRE  
CONFIG-SUBTUNNEL  
CONFIG-L2TPV3  
CONFIG-OPENVPN  
CONFIG-OPENVPN-SERVER
```

### Пример

```
esr(config-if-gi)# spanning-tree mst 1 cost 115
```



## spanning-tree mst max-hops

 В текущей версии ПО данная команда поддерживается только на маршрутизаторе ESR-1000.

Данной командой устанавливается максимальное количество транзитных участков для пакета BPDU, необходимых для формирования дерева и удержания информации о его строении. Если пакет уже прошел максимальное количество транзитных участков, то на следующем участке он отбрасывается. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
spanning-tree mst max-hops <NUM>
```

```
no spanning-tree mst max-hops
```

### Параметры

<NUM> – количество транзитных участков, задается в диапазоне [6..40].

### Значение по умолчанию

20

### Необходимый уровень привилегий

10


### Командный режим

CONFIG

### Пример

```
esr(config)# spanning-tree mst max-hops 10
```

## spanning-tree mst port-priority

 В текущей версии ПО данная команда поддерживается только на маршрутизаторе ESR-1000.

Данной командой устанавливается приоритет интерфейса для экземпляра MST. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
spanning-tree mst <INSTANCE> port-priority <PRIORITY>
```

```
no spanning-tree mst <INSTANCE> port-priority
```

### Параметры

<INSTANCE> – идентификатор MST-экземпляра, указывается в диапазоне [1..15];

<PRIORITY> – приоритет, указывается в диапазоне [0..240] с шагом 16.

### Значение по умолчанию

128

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-PORT-CHANNEL

### Пример

```
esr(config-if-gi)# spanning-tree port-priority 160
```

### spanning-tree pathcost method

Данной командой устанавливается метод определения ценности пути.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
spanning-tree pathcost method { short | long }
```

```
no spanning-tree pathcost method
```

### Параметры

long – значение ценности в диапазоне [1..200000000];

short – значение ценности в диапазоне [1..65535].

### Значение по умолчанию

short

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# spanning-tree pathcost method short
```

## spanning-tree portfast

Данной командой включается режим, в котором L2-интерфейс при поднятии на нем линка сразу переходит в состояние передачи, не дожидаясь истечения таймера.

Использование отрицательной формы команды (no) выключает режим моментального перехода в состояние передачи по поднятию линка.

### Синтаксис

```
[no] spanning-tree portfast
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключен.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-GRE

CONFIG-SUBTUNNEL

CONFIG-L2TPV3

CONFIG-OPENVPN

CONFIG-OPENVPN-SERVER

### Пример

```
esr(config-if-gi)# spanning-tree portfast
```

## spanning-tree port-priority

Данной командой устанавливается приоритет интерфейса в связующем дереве STP.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
spanning-tree port-priority <PRIORITY>  
no spanning-tree port-priority
```

**Параметры**

<PRIORITY> – приоритет, указывается в диапазоне [0..240] с шагом 16.

**Значение по умолчанию**

128

**Необходимый уровень привилегий**

10

**Командный режим**

```
CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-PORT-CHANNEL  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-GRE  
CONFIG-SUBTUNNEL  
CONFIG-L2TPV3  
CONFIG-OPENVPN  
CONFIG-OPENVPN-SERVER
```

**Пример**

```
esr(config-if-gi)# spanning-tree port-priority 160
```

**spanning-tree priority**

Данной командой настраивается приоритет связующего дерева STP.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
spanning-tree priority <PRIORITY>  
no spanning-tree priority
```

## Параметры

<PRIORITY> – приоритет, указывается в диапазоне [0..61440] с шагом 4096.

## Значение по умолчанию

32768

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

CONFIG-BRIDGE

## Пример

```
esr(config)# spanning-tree priority 4096
```

## Настройка и мониторинг VLAN

### force-up

Данная команда используется для включения режима активности VLAN вне зависимости от состояния интерфейсов, на которых разрешена обработка фреймов данного VLAN.

Использование отрицательной формы команды (no) отменяет режим активности VLAN.

### Синтаксис

[no] force-up

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-VLAN

## Пример

```
esr(config-vlan)# force-up
```

## ip internal-usage-vlan

**⚠** В текущей версии данная команда поддерживается только на маршрутизаторах ESR-1000/1200/1500/1511/1700.

Данная команда используется для резервирования VLAN для внутреннего использования на интерфейсе.

Использование отрицательной формы команды (no) отменяет резервирование.

## Синтаксис

```
ip internal-usage-vlan <VID>
```

```
no ip internal-usage-vlan
```

## Параметры

<VID> – идентификационный номер VLAN, задаётся в диапазоне [2...4094].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-FO

CONFIG-IF-PORT-CHANNEL

## Пример

```
esr(config-if-gi)# ip internal-usage-vlan 1500
```

## name

Данная команда используется для добавления описания VLAN.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
name <NAME>
```

```
no name
```

## Параметры

<NAME> – описание VLAN, задаётся строкой до 255 символов.

## Значение по умолчанию

Описание не задано.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-VLAN

## Пример

```
esr(config)# name L2-ACCESS
```

## show interfaces switch-port vlans

Данная команда используется для просмотра режима участия интерфейсов в VLAN.

## Синтаксис

```
show interfaces switch-port vlans [<IF>]
```

## Параметры

<IF> – имя физического или агрегированного интерфейса устройства, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

В команде можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будет отображена информация о всех интерфейсах заданной группы. При выполнении команды без параметра будет показана информация для всех физических интерфейсов.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
esr# show interfaces switch-port vlans gigabitethernet 1/0/1-7
Interface   PVID   Frame types   Ingress      Tagged      Untagged
-----
gi1/0/1     1      All           yes          101        1
gi1/0/2     1      All           yes          150-151    1
```

```

gil/0/3      1      All      yes      none      1
gil/0/4      1      All      yes      none      1
gil/0/5      1      All      yes      55       1
gil/0/6      1      All      yes      none      1
gil/0/7      1      All      yes      none      1
N/A - interface doesn't exist
N/S - interface is not a 802.1Q bridge port
ERR - can't get vlan setting for interface

```

## show vlans

Данная команда используется для просмотра информации об определенной VLAN.

### Синтаксис

```
show vlans [<VID>]
```

### Параметры

<VID> – идентификационный номер VLAN, диапазон допустимых значений [1 .. 4094].

Можно указать несколько VLAN перечислением через запятую «,» либо указать диапазон VLAN через дефис «-». При выполнении команды без параметра будут показаны все созданные VLAN.

### Необходимый уровень привилегий

1

### Командный режим

ROOT


### Пример

```

esr# show vlans
VID      Name              Tagged              Untagged
----      -
1        default          -----
2        --                -----

```

## show vlans internal-usage

 В текущей версии ПО данная команда поддерживается только на маршрутизаторах ESR-1000/1200/1500/1511/1700.

Данная команда используется для просмотра информации по VLAN, используемых системой.

### Синтаксис

```
show vlans internal-usage
```



## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

1

## Командный режим

ROOT


## Пример

```

esr# show vlans internal-usage
Usage      VID      Reserved  IP address
-----
gil/0/18   4088    No        Active
gil/0/16   4089    No        Active
gil/0/15   4090    No        Active

```

## switchport access vlan

 В текущей версии ПО данная команда поддерживается только на маршрутизаторах ESR-10/12V/12VF/20/21/100/200/3100.

Данная команда используется для включения/исключения интерфейса в/из VLAN в режиме работы access.

## Синтаксис

```
switchport access vlan <VID>
```

```
no switchport access vlan
```

## Параметры

<VID> – идентификационный номер VLAN, задаётся в диапазоне [2...4094].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-PORT-CHANNEL

## Пример

```
esr(config-if-gi)# switchport access vlan 50
```

### **switchport default-vlan tagged**

Данная команда используется для изменения членства интерфейса во VLAN по умолчанию на тегированное.

Использование отрицательной формы команды (no) изменяет членство интерфейса во VLAN по умолчанию на нетегированное.

### **Синтаксис**

```
[no] switchport default-vlan tagged
```

### **Параметры**

Команда не содержит параметров.

### **Необходимый уровень привилегий**

10

### **Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

## Пример

```
esr(config-if-gi)# switchport default-vlan tagged
```

### **switchport forbidden default-vlan**

Данная команда используется для удаления интерфейса из VLAN по умолчанию.

Использование отрицательной формы команды (no) разрешает добавление vlan на порту.

### **Синтаксис**

```
[no] switchport forbidden default-vlan
```

### **Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO


CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

**Пример**

```
esr(config-if-gi)# no switchport forbidden default-vlan
```

**switchport general acceptable-frame-type**

 В текущей версии ПО данная команда поддерживается только на маршрутизаторах ESR-1000/1200/1500/1511/1700.

Данной командой задается тип фреймов, которые может принимать интерфейс.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
switchport general acceptable-frame-type { tagged-only | all }
```

```
no switchport general acceptable-frame-type
```

**Параметры**

tagged-only – принимать только тегированные фреймы;

all – принимать все фреймы.

**Значение по умолчанию**

all

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-FO

## CONFIG-IF-PORT-CHANNEL

**Пример**

```
esr(config)# switchport general acceptable-frame-type tagged-only
```

**switchport general allowed vlan**

**!** В текущей версии ПО данная команда поддерживается только на маршрутизаторах ESR-1000/1200/1500/1511/1700.

Данная команда используется для включения/исключения интерфейса в/из VLAN.

**Синтаксис**

```
switchport general allowed vlan <ACT> <VID> [<TYPE>]
```

**Параметры**

<ACT> – назначаемое действие:

- add – включение интерфейса во VLAN;
- remove – исключение интерфейса из VLAN.

<VID> – идентификационный номер VLAN, задаётся в диапазоне [2...4094]. Можно задать диапазоном через «-» или перечислением через «,»;

<TYPE> – тип пакета:

- tagged – интерфейс будет передавать и принимать пакеты в указанных VLAN тегированными;
- untagged – интерфейс будет передавать пакеты в указанных VLAN нетегированными. VLAN, в которую будут направлены входящие нетегированные пакеты, настраивается командой *switchport general pvid*, описанной в [switchport general pvid](#).

**Значение по умолчанию**

Если не указывать параметр <TYPE>, то по умолчанию устанавливается «tagged».

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-FO

CONFIG-IF-PORT-CHANNEL

**Пример 1**

Исключить интерфейс из членства в VLAN 50:

```
esr(config-if-gi)# switchport general allowed vlan remove 50
```

## Пример 2

Включить интерфейс в VLAN 10-50 как тегированный:

```
esr(config-if-gi)# switchport general allowed vlan add 10-50
```

## switchport general allowed vlan auto-all

**⚠** В текущей версии ПО данная команда поддерживается только на маршрутизаторах ESR-1000/1200/1500/1511/1700.

Данная команда включает автоматическое добавление интерфейса во все созданные на маршрутизаторе VLAN.

Использование отрицательной формы команды (no) отключает автоматическое добавление порта во все созданные на маршрутизаторе VLAN.

## Синтаксис

```
[no] switchport general allowed vlan auto-all [ <TYPE> ]
```

## Параметры

<TYPE> – тип пакета:

- tagged – интерфейс будет передавать и принимать пакеты в указанных VLAN тегированными;
- untagged – интерфейс будет передавать пакеты в указанных VLAN нетегированными. VLAN, в которую будут направлены входящие нетегированные пакеты, настраивается командой *switchport general pvid*, описанной в [switchport general pvid](#).

## Значение по умолчанию

Отключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE


CONFIG-IF-F0

CONFIG-IF-PORT-CHANNEL

## Пример

```
esr(config-if-gi)# switchport general allowed vlan auto-all
```

## switchport general ingress-filtering disable

 В текущей версии ПО данная команда поддерживается только на маршрутизаторах ESR-1000/1200/1500/1511/1700.

Данная команда используется для выключения фильтрации входящих пакетов на основе присвоенного им значения VLAN ID.

Использование отрицательной формы команды (no) включает фильтрацию.

### Синтаксис

```
switchport general ingress-filtering disable
```

```
no switchport general ingress-filtering
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Фильтрация включена.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE


CONFIG-IF-FO

CONFIG-IF-PORT-CHANNEL

### Пример

```
esr(config-if-gi)# switchport general ingress-filtering disable
```

## switchport general macs-group

 В текущей версии ПО данная команда поддерживается только на маршрутизаторах ESR-1000/1200/1500/1511/1700.

Данной командой назначается профиль MAC-адресов и сопоставляется VLAN-ID, в который будут попадать пакеты с MAC-адресом источника, являющегося частью профиля MAC-адресов. Информация о конфигурировании профилей находится в разделе [Управление профилями](#).

Использование отрицательной формы команды (no) удаляет сопоставление профиля MAC-адресов и VLAN-ID.

## Синтаксис

```
switchport general macs-group <NAME> vlan <VID>  
no switchport general macs-group <NAME>
```

## Параметры

<NAME> – имя профиля MAC-адресов, задается строкой до 31 символа;  
<VID> – идентификационный номер VLAN, задается в диапазоне [2..4094].

## Необходимый уровень привилегий

10


## Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-FO  
CONFIG-IF-PORT-CHANNEL

## Пример

```
esr(config-if-gi)# switchport general macs-group OGM1 vlan 999
```

## switchport general pvid

 В текущей версии ПО данная команда поддерживается только на маршрутизаторах ESR-1000/1200/1500/1511/1700.

Данной командой устанавливается идентификатор VLAN-порта (PVID) для входящего нетегированного трафика.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
switchport general pvid <VID>
no switchport general pvid
```

## Параметры

<VID> – идентификационный номер VLAN, задаётся в диапазоне [1...4094].

## Значение по умолчанию

1

## Необходимый уровень привилегий

10


## Командный режим

```
CONFIG-IF-GI
CONFIG-IF-TE
CONFIG-IF-FO
CONFIG-IF-PORT-CHANNEL
```

## Пример

```
esr(config-if-gi)# switchport general pvid 999
```

## switchport macs-group

 В текущей версии ПО данная команда поддерживается только на маршрутизаторах ESR-10/12V/12VF/15/15R/15VF/20/21/30/31/100/200/3100/3200/3200L/3300.

Данной командой назначается профиль MAC-адресов и сопоставляется VLAN-ID, в который будут попадать пакеты с MAC-адресом источника, являющегося частью профиля MAC-адресов. Информация о конфигурировании профилей находится в разделе [Управление профилями](#).

Использование отрицательной формы команды (no) удаляет сопоставление профиля MAC-адресов и VLAN-ID.

## Синтаксис

```
switchport macs-group <NAME> vlan <VID>
no switchport macs-group <NAME>
```

## Параметры

<NAME> – имя профиля MAC-адресов, задается строкой до 31 символа;

<VID> – идентификационный номер VLAN, задаётся в диапазоне [2..4094].



**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

**Пример**

```
esr(config-if-gi)# switchport macs-group OGM1 vlan 999
```

**switchport trunk allowed vlan**

**⚠** В текущей версии ПО данная команда поддерживается только на маршрутизаторах ESR-10/12V/12VF/15/15R/15VF/20/21/30/31/100/200/3100/3200/3200L/3300.

Данная команда используется для включения/исключения интерфейса в/из VLAN в режиме работы trunk.

**Синтаксис**

```
switchport trunk allowed vlan <ACT> <VID>
```

**Параметры**

<ACT> – назначаемое действие:

- add – включение интерфейса во VLAN;
- remove – исключение интерфейса из VLAN.

<VID> – идентификационный номер VLAN, задаётся в диапазоне [2...4094]. Можно задать диапазоном через «-» или перечислением через «,».

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO


CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

### Пример

```
esr(config-if-gi)# switchport trunk allowed vlan add 10-50
```

### switchport trunk allowed vlan auto-all

 В текущей версии ПО данная команда поддерживается только на маршрутизаторах ESR-10/12V/12VF/15/15R/15VF/20/21/30/31/100/200/3100/3200/3200L/3300.

Данная команда включает автоматическое добавление порта во все созданные на маршрутизаторе VLAN.

Использование отрицательной формы команды (no) отключает автоматическое добавление порта во все созданные на маршрутизаторе VLAN.

### Синтаксис

```
[no] switchport trunk allowed vlan auto-all
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

### Пример

```
esr(config-if-gi)# switchport trunk allowed vlan auto-all
```

## switchport trunk native-vlan

**⚠** В текущей версии ПО данная команда поддерживается только на маршрутизаторах ESR-10/12V/12VF/15/15R/15VF/20/21/30/31/100/200/3100/3200/3200L/3300.

Данная команда используется для настройки VLAN по умолчанию для интерфейса в режиме работы trunk. Весь нетегированный трафик, поступающий на данный интерфейс, направляется в данную VLAN.

### Синтаксис

```
switchport trunk native-vlan <VID>
no switchport trunk native-vlan
```

### Параметры

<VID> – идентификационный номер VLAN, задаётся в диапазоне [2..4094].

### Значение по умолчанию

1

### Необходимый уровень привилегий

10

### Командный режим

```
CONFIG-IF-GI
CONFIG-IF-TE
CONFIG-IF-TWE
CONFIG-IF-FO
CONFIG-IF-HU
CONFIG-IF-PORT-CHANNEL
```

### Пример

```
esr(config-if-gi)# switchport trunk native-vlan 55
```

## vlan

Данной командой добавляется VLAN в систему и осуществляется переход в режим настройки параметров VLAN. На маршрутизаторе всегда существует VLAN с идентификатором 1, все интерфейсы по умолчанию включены в данный VLAN.

Использование отрицательной формы команды (no) удаляет VLAN.

### Синтаксис

```
[no] vlan <VID>
```

## Параметры

<VID> – идентификационный номер VLAN, задаётся в диапазоне [2...4094].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# vlan 40
```

## Настройка и мониторинг Voice-VLAN

### application

Данной командой настраивается приложение, которое будет числиться в пакете LLDP-MED при рассылке LLDPDU с интерфейсов, на которых назначена сетевая политика (network-policy).

Использование отрицательной формы команды (no) удаляет настройку приложения для политики.

### Синтаксис

```
application <APP-TYPE>
```

```
no application
```

### Параметры

<APP-TYPE> – тип приложения, для которого будет срабатывать network-policy. Принимает значения:

- voice;
- voice-signaling;
- guest-voice;
- guest-voice-signaling;
- softphone-voice;
- video-conferencing;
- streaming-video;
- video-signaling.

### Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-NET-POLICY

### Пример

```
esr(config-net-policy)# application voice
```

### dscp

Данной командой настраивается значение DSCP, которое будет числиться в пакете LLDP-MED при рассылке LLDPDU с интерфейсов, на которых назначена сетевая политика (network-policy).

Использование отрицательной формы команды (no) удаляет рассылку DSCP для данной сетевой политики (network-policy).

### Синтаксис

```
dscp <DSCP>
```

```
dscp
```

### Параметры

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

### Значение по умолчанию

Рассылка DSCP в LLDPDU-MED отключена.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-NET-POLICY


### Пример

```
esr(config-net-policy)# dscp 62
```

### lldp network-policy

Данной командой на интерфейс назначается созданная ранее сетевая политика (network-policy).

Использование отрицательной формы команды (no) снимает назначенную ранее сетевую политику с интерфейса.

 LLDPDU-сообщения, описанные политикой, будут рассылаться только если в глобальной конфигурации включена поддержка LLDP-MED (см. команду **lldp med fast-start enable** в разделе [Настройка протокола LLDP](#)).

## Синтаксис

```
[no] lldp network-policy <NAME>
```

## Параметры

<NAME> – имя назначаемой политики, задается строкой до 31 символа.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

## Пример

```
esr(config-if-gi) lldp network-policy ip-phones
```

## network-policy

Данной командой создается сетевая политика (network-policy) и осуществляется переход в режим настройки параметров политики.

Использование отрицательной формы команды (no) удаляет созданную ранее сетевую политику (network-policy).

## Синтаксис

```
[no] network-policy <NAME>
```

## Параметры

<NAME> – имя создаваемой политики, задается строкой до 31 символа.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# network-policy ip-phones
```

**priority**

Данной командой настраивается значение COS, которое будет числиться в пакете LLDP-MED при рассылке LLDPDU с интерфейсов, на которых назначена сетевая политика (network-policy).

Использование отрицательной формы команды (no) удаляет рассылку COS для данной сетевой политики (network-policy).

**Синтаксис**

```
priority <COS>
```

```
no priority
```

**Параметры**

<COS> – значение приоритета, принимает значения:

- best-effort – COS0;
- background – COS1;
- excellent-effort – COS2;
- critical-applications – COS3;
- video – COS4;
- voice – COS5;
- internetwork-control – COS6;
- network-control – COS7.

**Значение по умолчанию**

Рассылка COS в LLDPDU-MED отключена.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-NET-POLICY

**Пример**

```
esr(config-net-policy)# priority voice
```

**vlan**

Данной командой настраивается значение VLAN, которое будет числиться в пакете LLDP-MED при рассылке LLDPDU с интерфейсов, на которых назначена сетевая политика (network-policy).

Использование отрицательной формы команды (no) удаляет конфигурацию VLAN для данной сетевой политики (network-policy).

**Синтаксис**

```
vlan <VID> [tagged]
```

**Параметры**

<VID> – идентификационный номер VLAN, принимает значения [1...4094];

tagged – ключ, при установке которого абонентское устройство будет отправлять Ethernet-фреймы указанного приложения в тегированном виде.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-NET-POLICY

**Пример**

```
esr(config-net-policy)# vlan 3258 tagged
```



## 16 Работа с адресными таблицами

- `clear arp-cache`
- `clear ipv6 neighbors`
- `clear mac address-table`
- `ip arp`
- `ip arp inspection`
- `ip arp reachable-time`
- `ipv6 nd`
- `ipv6 nd reachable-time`
- `logging firewall arp-inspection`
- `mac address-table aging-time`
- `mac address-table save-secure-freq`
- `port-security max`
- `port-security mode`
- `port-security unknown-sa-action`
- `show arp`
- `show arp configuration`
- `show ipv6 neighbors`
- `show ipv6 neighbors configuration`
- `show mac address-table`

### clear arp-cache

Команда используется для очистки содержимого ARP-таблицы.

#### Синтаксис

```
clear arp-cache [ <OPTIONS> ]
```

#### Параметры

<OPTIONS> – параметры команды для детализации запрашиваемой информации, опциональный параметр:

- `vrf <VRF>` – имя экземпляра VRF, задается строкой до 31 символа. Опциональный параметр, при указании которого будет очищена ARP-таблица в указанном VRF;
- `<IF>` – имя интерфейса устройства, задается в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);
- `ip-address <ADDR>` – IP-адрес, по которому ведется поиск, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];
- `mac-address <ADDR>` – MAC-адрес, по которому ведется поиск, задается в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF].

#### Необходимый уровень привилегий

10

#### Командный режим

ROOT

#### Пример

```
esr# clear arp-cache ip-address 10.0.0.8
```

## clear ipv6 neighbors

Команда используется для очистки содержимого IPv6 Neighbor Discovery таблицы.

### Синтаксис

```
clear ipv6 neighbors [<OPTIONS> ]
```

### Параметры

<OPTIONS> – параметры команды для детализации запрашиваемой информации, опциональный параметр:

- vrf <VRF> – имя экземпляра VRF, задается строкой до 31 символа. При указании данного параметра будет очищена IPv6 Neighbor Discovery таблица в указанном VRF;
- <IF> – имя интерфейса устройства, задается в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);
- ipv6-address <IPv6-ADDR> – указывается IPv6-адрес, по которому ведется поиск, задается в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];
- mac-address <ADDR> – MAC-адрес, по которому ведется поиск, задается в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF].

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример

```
esr# clear ipv6 neighbors
```

## clear mac address-table

Команда используется для удаления информации об изученных MAC-адресах.

### Синтаксис

```
clear mac address-table [ { dynamic | static } ] [ { interface { <IF> | host-port <U/S/P> } | bridge <BRIDGE-ID> | tunnel { gre | softgre } <ID> | vlan <VLAN-ID> }
```

### Параметры

<IF> – имя интерфейса устройства, задается в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<U/S/P> – юнит (1), слот (0) и номер интерфейса пакетного процессора;

<BRIDGE-ID> – идентификационный номер моста, задается в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<ID> – идентификатор туннеля;

<VLAN-ID> – номер vlan. Возможно задание списка vlan через символ ",", без пробелов, диапазона vlan через символ "-" и/или комбинации списков и диапазонов.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример

```
esr# clear mac address-table
```

### ip arp

Данной командой добавляется статическая запись в ARP-таблицу.

Использование отрицательной формы команды (no) удаляет статическую запись из ARP-таблицы.

### Синтаксис

```
ip arp [ vrf <VRF> ] <IP> <MAC> { <IF> | <TUN> }
```

```
no ip arp [ vrf <VRF> ] <IP>
```

### Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа;

<IP> – IP-адрес статической записи, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<MAC> – MAC-адрес клиента, которому будет выдан IPv6-адрес, задается в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF];

<IF> – имя интерфейса устройства, задается в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<TUN> – имя туннеля устройства, задается в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#).

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# ip arp 192.168.54.22 a8:f9:4b:ab:2e:d0 bridge 3
```

## ip arp inspection

Данной командой включается функция контроля протокола ARP (ARP Inspection), предназначенная для защиты от атак с использованием протокола ARP (например, ARP-spoofing — перехват ARP-трафика).

С включенной функцией контроля ARP при приеме ARP-ответа производится сравнение старого и нового MAC-адресов, и при обнаружении его изменения запускается процедура верификации. Посылается ARP-запрос, требующий всем хозяевам IP-адреса сообщить свои MAC-адреса. Если выполняется атака, настоящая система, имеющая этот IP-адрес, ответит на запрос, и таким образом атака будет распознана. Если же изменение MAC-адреса было связано не с атакой, а со стандартными ситуациями, ответа, содержащего "старый" MAC-адрес, не будет, и по прошествии определенного таймаута система обновит запись в кэше.

Использование отрицательной формы команды (no) выключает функцию контроля.

### Синтаксис

```
[no] ip arp inspection
```

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-LOOPBACK

CONFIG-BRIDGE

### Пример

```
esr(config-if-gi)# ip arp inspection
```

## ip arp reachable-time

Данной командой устанавливается время жизни записи в ARP-таблице.

Использование отрицательной формы команды (no) устанавливает значение параметра arp reachable-time по умолчанию.

**Синтаксис**

```
ip arp reachable-time <TIME>
no ip arp reachable-time
```

**Параметры**

< TIME > – время жизни динамических MAC-адресов, в миллисекундах. Допустимые значения от 5000 до 100000000 миллисекунд. Реальное время обновления записи варьируется от  $[0,5;1,5] * < TIME >$ .

**Необходимый уровень привилегий**

10

**Значение по умолчанию**

160000

**Командный режим**

```
CONFIG
CONFIG-IF-GI
CONFIG-IF-TE
CONFIG-IF-TWE
CONFIG-IF-FO
CONFIG-IF-HU
CONFIG-IF-OOB
CONFIG-IF-SUB
CONFIG-IF-QINQ
CONFIG-IF-PORT-CHANNEL
CONFIG-LOOPBACK
CONFIG-BRIDGE
```

**Пример**

```
esr(config-if-gi)# ip arp reachable-time 6000
```

**ipv6 nd**

Данной командой добавляется статическая запись в ND-таблицу.

Использование отрицательной формы команды (no) удаляет статическую запись из ND-таблицы.

**Синтаксис**

```
ipv6 nd [ vrf <VRF> ] <IPV6> <MAC> {<IF> | <TUN>}
no ipv6 nd [ vrf <VRF> ] <IP>
```

## Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа;

<IPv6-ADDR> – указывается IPv6-адрес, задается в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<MAC> – MAC-адрес клиента, которому будет выдан IPv6-адрес, задается в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF];

<IF> – имя интерфейса устройства, задается в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<TUN> – имя туннеля устройства, задается в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#).

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# ipv6 a8:f9:4b:ab:2e:d0
```

## ipv6 nd reachable-time

Данной командой устанавливается время, в течение которого удаленный узел IPv6 считается доступным при отсутствии активности узла.

Использование отрицательной формы команды (no) устанавливает значение параметра nd reachable-time по умолчанию.

## Синтаксис

```
ipv6 nd reachable-time <TIME>
```

```
no ipv6 nd arp reachable-time
```

## Параметры

<TIME> – время жизни записи об удаленном узле IPv6 в таблице ND протокола, в миллисекундах. Допустимые значения от 5000 до 100000000 миллисекунд. Реальное время обновления записи варьируется от [0,5;1,5]\*< TIME >.

## Значение по умолчанию

30000

## Необходимый уровень привилегий

10

**Командный режим**

CONFIG

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-LOOPBACK

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

**Пример**

```
esr(config-if-gi)# ipv6 nd reachable-time 27000
```

**logging firewall arp-inspection**

Команда включает режим логирования атак с использованием протокола ARP, работает только совместно с командой **ip arp inspection**.

При включенных командах **ip arp inspection** и **logging firewall arp-inspection** в случае обнаружения атаки типа ARP-Spoofing в Syslog будет отправлено сообщение с указанием интерфейса существующей и подменяемой arp-записях.

Использование отрицательной формы команды (no) отключает режим логирования.

**Синтаксис**

```
[no] logging firewall arp-inspection
```

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# logging firewall arp-inspection
```

```
2024-08-13T12:29:01+07:00 %FIREWALL-W-WARN: arp-inspection a8:f9:4b:ac:4d:3e(192.168.1.10) ->
a8:f9:4b:aa:cc:a4(192.168.1.11) denied (gi1/0/1 08:00:27:d2:26:79)
```

## mac address-table aging-time

Командой устанавливается время жизни динамических MAC-адресов в forwarding-таблице.

Использование отрицательной формы команды (no) устанавливает значение «aging time» по умолчанию.

### Синтаксис

```
mac address-table aging-time <AGING TIME>
```

```
[no] mac address-table aging time
```

### Параметры

<AGING TIME> – время жизни динамических MAC-адресов, в секундах. Допустимые значения:

- ESR-1000/1200/1500/1511/1700 – от 10 до 630 секунд. При значении 0 таймер выключен.
- ESR-10/12V/12VF/15/15R/15VF/20/21/30/31/100/200 – от 20 до 630 секунд.

### Значение по умолчанию

300

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# mac address-table aging-time 30
```

## mac address-table save-secure-freq

 В текущей версии данная команда поддерживается только на маршрутизаторах ESR-1000.

Данной командой устанавливается частота сохранения списка статических (secure) MAC-адресов.

Использование отрицательной формы команды (no) устанавливает значение «mac address-table save-secure-freq» по умолчанию.

### Синтаксис

```
mac address-table save-secure-freq <SAVE-SECURE-FREQ>
```

```
[no] mac address-table save-secure-freq
```

### Параметры

<SAVE-SECURE-FREQ> – частота сохранения списка статических (secure) MAC-адресов, принимает значение [600..86400] секунд.



**Значение по умолчанию**

1200 секунд

**Необходимый уровень привилегий**

15


**Командный режим**

CONFIG

**Пример**

```
esr(config)# mac address-table save-secure-freq 650
```

**port-security max**

 В текущей версии ПО данная функция поддерживается только на маршрутизаторе ESR-1000.

Данной командой устанавливается максимальное количество MAC-адресов, разрешенных для запоминания на порту.

Использование отрицательной формы команды (no) отключает «port-security».

**Синтаксис**

```
port-security max <MAX>
```

```
no port-security max
```

**Параметры**

<MAX> – максимальное количество MAC-адресов, которое будет запоминаться портом, принимает значения [1..1024].

**Необходимый уровень привилегий**

15

**Командный режим**


CONFIG-IF-GI

CONFIG-TE

**Пример**

```
esr(config-if-gi)# port-security max 1
```

## port-security mode

 В текущей версии ПО данная функция поддерживается только на маршрутизаторе ESR-1000.

Данная команда позволяет настроить режим «port-security».

Использование отрицательной формы команды (no) отключает режим безопасности.

### Синтаксис

```
port-security mode [<OPTIONS>]
```

```
no port-security mode
```

### Параметры

<OPTIONS> – параметры команды для выбора режима port-security:

limited – при включении данного режима:

- с порта удаляются все выученные MAC-адреса;
- количество адресов, которое снова может запомнить порт, ограничивается текущей конфигурацией;
- MAC-адреса не сохраняются между аппаратными перезагрузками;
- время хранения MAC-адресов зависит от времени жизни динамических MAC-адресов в forwarding-таблице.

lock – при включении данного режима:

- на порту сохраняются все выученные MAC-адреса;
- порт не запоминает новые адреса;
- MAC-адреса сохраняются между аппаратными перезагрузками;
- время хранения MAC-адресов зависит от времени жизни динамических MAC-адресов в forwarding-таблице.

secure-delete-on-reset – при включении данного режима:

- с порта удаляются все выученные MAC-адреса;
- количество адресов, которое снова может запомнить порт, ограничивается текущей конфигурацией;
- MAC-адреса не сохраняются между аппаратными перезагрузками;
- время хранения MAC-адресов не зависит от времени жизни динамических MAC-адресов в forwarding-таблице.

secure-permanent – при включении данного режима:

- с порта удаляются все выученные MAC-адреса;
- количество адресов, которое снова может запомнить порт, ограничивается текущей конфигурацией;
- MAC-адреса сохраняются между аппаратными перезагрузками;
- время хранения MAC-адресов не зависит от времени жизни динамических MAC-адресов в forwarding-таблице.

### Необходимый уровень привилегий

15

### Командный режим


CONFIG-IF-GI

## CONFIG-TE

### Пример

```
esr(config-if-gi)# port-security mode secure-delete-on-reset  
esr(config-if-gi)# port-security mode secure-permanent
```

### port-security unknown-sa-action

 В текущей версии ПО данная функция поддерживается только на маршрутизаторе ESR-1000.

Командой устанавливается запрет на передачу пакетов с неизвестными MAC-адресами.

Использование отрицательной формы команды (no) разрешает передачу пакетов с неизвестными MAC-адресами.

### Синтаксис

```
port-security unknown-sa-action discard  
no port-security unknown-sa-action
```

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IF-GI  
CONFIG-TE

### Пример

```
esr(config-if-gi)# port-security unknown-sa-action discard
```

### show arp

Команда используется для просмотра ARP-таблицы.

### Синтаксис

```
show arp [<OPTIONS>]
```

### Параметры

<options> – параметры команды для детализации запрашиваемой информации, опциональный параметр:

- vrf <VRF> – имя экземпляра VRF, задается строкой до 31 символа. При указании данного параметра будет отображена ARP-таблица в указанном VRF;
- <IF> – наименование системного интерфейса или списка интерфейсов, задается в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#). Отображается только информация по указанным интерфейсам;

- <TUN> – наименования туннелей, задаются в виде, описанном в разделе [Типы и порядок именованя туннелей маршрутизатора](#).
- mac-address <MAC> – MAC-адрес, по которому ведется поиск, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF];
- ip-address <ADDR> – IP-адрес, по которому ведется поиск, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
esr# show arp
Interface          IP address      MAC address      State            Age(min)
-----
bridge 1           192.168.1.1    a8:f9:4b:aa:00:40 --                --
gi1/0/5            10.255.100.1   d8:50:e6:d2:f0:46 reachable        2
gi1/0/5            10.255.100.5   a8:f9:4b:aa:00:45 --                --
```

### show arp configuration

Команда используется для просмотра значений времени жизни записей в ARP-таблице.

### Синтаксис

```
show arp configuration <IF>
```

### Параметры

<IF> – наименования системных интерфейсов, задаются в виде, описанном в разделе [Типы и порядок именованя интерфейсов маршрутизатора](#).

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
esr# sh arp configuration gigabitethernet 1/0/1-5
Globally configured ARP reachable time is 6000 msec
Interface          ARP reachable time, msec
-----
gi1/0/1            6000
gi1/0/2            6000
```

```

gil/0/3          6000
gil/0/4          6000
gil/0/4          6000

```

## show ipv6 neighbors

Команда используется для просмотра IPv6 Neighbor Discovery таблицы.

### Синтаксис

```
show ipv6 neighbors [<OPTIONS>]
```

### Параметры

<OPTIONS> – параметры команды для детализации запрашиваемой информации, опциональный параметр:

- vrf <VRF> – имя экземпляра VRF, задается строкой до 31 символа. При указании данного параметра будет отображена IPv6 Neighbor Discovery таблица в указанном VRF;
- <IF> – наименование системного интерфейса или списка интерфейсов, задается в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#). Отображается только информация по указанным интерфейсам;
- mac-address <MAC> – указывается MAC-адрес, по которому ведется поиск, задается в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF];
- ipv6-address <IPV6-ADDR> – указывается IPv6-адрес, по которому ведется поиск, задается в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```

esr# show ipv6 neighbors
Interface          IPv6 address          MAC address           State                Age(min)
-----
gil/0/5            fc00::1               d8:50:e6:d2:f0:46    reachable           1
gil/0/5            fc00::2               a8:f9:4b:aa:00:45    --                  --
bridge 1           fe80::aaf9:4bff:feaa:40 a8:f9:4b:aa:00:40    --                  --
bridge 2           fe80::aaf9:4bff:feaa:40 a8:f9:4b:aa:00:40    --                  --
gil/0/5            fe80::aaf9:4bff:feaa:45 a8:f9:4b:aa:00:45    --                  --
gil/0/5            ff02::16              33:33:00:00:00:16    norarp              --
gil/0/5            ff02::fb              33:33:00:00:00:fb    norarp              --
gil/0/5            ff02::1:ff00:1        33:33:ff:00:00:01    norarp              --
gil/0/5            ff02::1:ff00:2        33:33:ff:00:00:02    norarp              --

```

## show ipv6 neighbors configuration

Команда используется для просмотра значений времени жизни записи об удаленном узле в таблице ND протокола.

## Синтаксис

```
show ipv6 neighbors configuration <IF>
```

## Параметры

<IF> – наименования системных интерфейсов, задаются в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```

esr# sh ipv6 neighbors configuration tengigabitethernet 1/0/1-2
Globally configured NDP reachable time is 30000 msec
Interface          ND reachable time, msec
-----
te1/0/1            30000
te1/0/2            30000

```

## show mac address-table

Команда используется для просмотра информации, находящейся в таблице MAC-адресов.

## Синтаксис

```
show mac address-table [ { count { bridge <BRIDGE-ID> | l2vpn p2p <P2P-NAME> } | { static | dynamic } { bridge <BRIDGE-ID> | l2vpn p2p <P2P-NAME> | interface { <IF> | host-port <U/S/P> } | tunnel { gre | softgre } <ID> | vlan <VLAN-ID> | mac <MAC-ADDR> <MAC-MASK> } } ]
```

## Параметры

- count – просмотр количества записей в таблице MAC-адресов. Список MAC-адресов не отображается;
- static – просмотр записей в таблице MAC-адресов внесенных из конфигурации маршрутизатора;
- dynamic – просмотр записей в таблице MAC-адресов внесенных в процессе изучения MAC-адресов;
- <BRIDGE-ID> – идентификационный номер моста, задается в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);
- <P2P-NAME> – имя существующего p2p-сервиса, задается строкой до 31 символа;
- <IF> – имя интерфейса устройства, задается в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);
- <HOST-U/S/P> – интерфейс пакетного процессора. Указывается в виде юнит (1), слот (0) и номер интерфейса пакетного процессора;
- <ID> – идентификатор туннеля, задается в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#);
- <VLAN-ID> – номер vlan. Возможно задание списка vlan через символ "," без пробелов, диапазона vlan через символ "-" и/или комбинации списков и диапазонов;

- <MAC-ADDR> – MAC-адрес, по которому ведётся поиск, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF];
- [MAC-MASK] – маска MAC-адреса, опциональный параметр, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF]. Биты маски, установленные в 0, задают биты MAC-адреса, исключаемые из сравнения при поиске. Значение маски по умолчанию FF:FF:FF:FF:FF:FF.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример

```
esr# show mac address-table
VID      MAC Address      Interface          Type
-----  -
102      a8:f9:4b:aa:44:bb host-port 1/0/2             Dynamic
101      a8:f9:4b:aa:44:bb host-port 1/0/2             Dynamic
100      a8:f9:4b:aa:44:bb host-port 1/0/2             Dynamic
3 valid mac entries
```

## 17 Настройка VRF

- [description](#)
- [ip source-vrf](#)
- [ip vrf](#)
- [ip vrf forwarding](#)
- [show ip vrf](#)

### description

Данная команда используется для назначения описания конфигурируемому экземпляру VRF. Использование отрицательной формы команды (no) удаляет установленное описание.

### Синтаксис

```
description <DESCRIPTION>  
no description
```

### Параметры

<DESCRIPTION> – описание экземпляра VRF, задаётся строкой до 255 символов.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-VRF

### Пример

```
esr(config-vrf)# description "VRF1"
```

### ip source-vrf

Данной командой осуществляется перевод системы копирования по TFTP, SCP, FTP, SFTP в VRF. Использование отрицательной формы команды (no) системы копирования в стандартный режим.

### Синтаксис

```
ip { tftp | ftp | ssh | sftp | http } source-vrf <VRF>  
no ip { tftp | ftp | ssh | sftp | http } source-vrf
```

### Параметры

<VRF> – имя VRF, задается строкой до 31 символа.

tftp – копирование по протоколу TFTP;

ftp – копирование по протоколу FTP;

ssh – копирование по протоколу SCP;



sftp – копирование по протоколу SFTP;

http – копирование по протоколу HTTP.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# ip tftp source-vrf vrf1
```

### ip vrf

Данной командой в системе создается экземпляр VRF и осуществляется переход в режим настройки параметров экземпляра VRF.

Использование отрицательной формы команды (no) удаляет экземпляр VRF из системы.

### Синтаксис

```
[no] ip vrf <VRF>  
no ip vrf all
```

### Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# ip vrf VRF1  
esr(config-vrf)#
```

### ip vrf forwarding

Данной командой задаётся имя экземпляра VRF, в котором будут использоваться указанные сетевой интерфейс, мост, зона безопасности, сервер динамической авторизации (DAS) или группа правил NAT.

Использование отрицательной формы команды (no) удаляет привязку сетевого интерфейса, моста, зоны безопасности или группы правил NAT к экземпляру VRF.

**Синтаксис**

```
ip vrf forwarding <VRF>  
no ip vrf forwarding
```

**Параметры**

<VRF> – имя VRF, задается строкой до 31 символа.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-OOB  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-CELLULAR-MODEM  
CONFIG-BRIDGE  
CONFIG-LOOPBACK  
CONFIG-VTI  
CONFIG-GRE  
CONFIG-LT  
CONFIG-SUBTUNNEL  
CONFIG-ZONE  
CONFIG-DNAT-RULESET  
CONFIG-SNAT-RULESET  
CONFIG-DAS-SERVER  
CONFIG-SNMP-USER  
CONFIG-SNMP-VIEW  
CONFIG-L2TP  
CONFIG-PPPOE  
CONFIG-PPTP  
CONFIG-OPENVPN  
CONFIG-IPSEC-VPN  
CONFIG-WIREGUARD-SERVER

CONFIG-WIREGUARD-TUNNEL-PEER

CONFIG-IF-E1

CONFIG-IF-MULTILINK

### Пример

```
esr(config-snat-ruleset)# ip vrf forwarding VRF1
```

### show ip vrf

Команда используется для просмотра информации о существующих в системе экземплярах VRF.

### Синтаксис

```
show ip vrf [ <VRF> ]
```

### Параметры

<VRF> – имя VRF, задается строкой до 31 символа.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
esr# show ip vrf
Name                               Interfaces
-----                               -
VRF1                                gi1/0/8
VRF2                                gi1/0/10.22
test                                 gi1/0/2
```

## 18 Настройка IP-адресации

- [ip address](#)
- [ip redirects](#)
- [ip route source-route](#)
- [ip unnumbered](#)
- [ip unreachable](#)
- [show ip interfaces](#)

### ip address

Данной командой создаётся IP-интерфейс и добавляются IP-адрес и маска подсети для конфигурируемого интерфейса (физического интерфейса, группы агрегации каналов, туннеля или сетевого моста).

Использование отрицательной формы команды (no) удаляет IP-адрес с интерфейса. При удалении последнего адреса IP-интерфейс уничтожается.

**!** При создании IP-интерфейса система резервирует наибольший незанятый VLAN ID, который будет использоваться внутри системы. Для каждого IP-интерфейса на Ethernet-порту резервируется VLAN. Если VLAN уже был зарезервирован для IPv6-интерфейса, то для IP-интерфейса VLAN резервироваться не будет. Можно зарезервировать VLAN ID для внутреннего использования явно с помощью команды `ip internal-usage vlan <VLAN_ID>`.

**!** На устройствах ESR-1x сеть 192.0.2.0/24 зарезервирована под служебный функционал.

### Синтаксис

```
ip address <ADDR/LEN> [ secondary ] [ unit <ID> ]
no ip address { <ADDR/LEN> [ unit <ID> ] | all }
```

### Параметры

<ADDR/LEN> – IP-адрес и длина маски подсети, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].

<ID> – номер юнита, принимает значения [1..2];

secondary – ключ указывает, что настроенный адрес является дополнительным IP-адресом. Если это ключевое слово отсутствует, настроенный адрес является основным IP-адресом. Возможно указать до 7 дополнительных IP-адресов.

all – команда удаляет все IP-адреса на интерфейсе.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU  
CONFIG-IF-OOB  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-SERIAL  
CONFIG-IF-PORT-CHANNEL  
CONFIG-CELLULAR-MODEM  
CONFIG-IF-E1  
CONFIG-IF-MULTILINK  
CONFIG-VTI  
CONFIG-GRE  
CONFIG-IP4IP4  
CONFIG-BRIDGE  
CONFIG-LOOPBACK  
CONFIG-LT  
CONFIG-WIREGUARD

### Пример

```
esr(config-if-gi)# ip address 192.168.25.25/24
```

### ip redirects

Данной командой включается механизм отправки ICMP-сообщений о существовании более приоритетного маршрутизатора в данной IP-сети для конкретного IP-назначения.

Использование отрицательной формы команды (no) отключает механизм отправки ICMP-сообщений о существовании более приоритетного маршрутизатора в данной IP-сети для конкретного IP-назначения.

### Синтаксис

```
[no] ip redirects
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

15

### Значение по умолчанию

Отправка ICMP-сообщений о существовании более приоритетного маршрутизатора в данной IP-сети для конкретного IP-назначения включена.

**Командный режим**

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-OOB  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-LOOPBACK  
CONFIG-IF-E1  
CONFIG-IF-MULTILINK  
CONFIG-GRE  
CONFIG-IP4IP4  
CONFIG-LT

**Пример**

```
esr(config-if-gi)# no ip redirects
```

**ip route source-route**

Данной командой на маршрутизаторе включается поддержка опции ip source-route.

Использование отрицательной формы команды (no) поддержка опции ip source-route отключается.

**Синтаксис**

```
[no] ip route source-route
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

## Командный режим

CONFIG

### Пример

```
esr(config)# ip route source-route
```

## ip unnumbered

Данной командой включается режим работы интерфейса с использованием IP-адреса, назначенного на другой интерфейс.

Использование отрицательной формы команды (no) отключает режим ip unnumbered.

### Синтаксис

```
ip unnumbered { <IF> | <TUN> }
```

```
no ip unnumbered
```

### Параметры

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<TUN> – наименования туннелей, задаются в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#).

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-GRE

CONFIG-IP4IP4

## Пример

```
esr(config-if-e1)# ip unnumbered gigabitethernet 1/0/1
esr(config-if-e1)#
```

## ip unreachable

Данной командой включается отправка ICMP-пакетов о недоступности конечного адреса.

Использование отрицательной формы команды (no) отключает возможность отправки ICMP-пакетов о недоступности конечного адреса.

## Синтаксис

```
[no] ip unreachable
```

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

15

## Значение по умолчанию

Отправка ICMP-пакетов о недоступности конечного адреса включена.

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT



## Пример

```
esr(config-if-gi)# no ip unreachablees
```

## show ip interfaces

Команда используется для просмотра информации о существующих в системе IP-интерфейсах.

### Синтаксис

```
show ip interfaces [ { <IF> | <TUN> | vrf <VRF> [ ip-address <ADDR> ] } ]
```

### Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа. При указании данного параметра будут отображены IP-интерфейсы в указанном VRF;

<ADDR> – IP-адрес, задается в виде AAA.BBB.CCC.DDD/EE, где каждая часть принимает значения [0..255]. При указании данного параметра будет отображен IP-интерфейс с указанным IP-адресом;

<IF> – наименования системных интерфейсов, задаются в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<TUN> – наименования туннелей, задаются в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#).

В команде можно указать несколько системных интерфейсов. Если не указывать индексы интерфейсов, то будут отображены все IP-интерфейсы, относящиеся к системным интерфейсам указанного типа.

Если в команде указан определенный системный интерфейс, получающий IP-параметры по протоколу DHCP, то будут отображены настройки DHCP-клиента и состояние текущей аренды IP-параметров.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

## Пример

```
esr# show ip interfaces
IP address          Interface          Admin  Link
Type      Precedence
-----
-----
155.0.0.60/24      br10              Up     Up
static primary
16.0.0.2/24       gi1/0/16          Up     Up
DHCP              --
10.0.0.1/8        gi1/0/18          Up     Up
static primary
180.0.0.1/24      gi1/0/18          Up     Up
static secondary
```

```
192.168.1.1/24
static primary
25.0.0.2/30
static primary
10.1.0.2/24
static primary
```

```
gi1/0/2      Up      Up
gre 25       Up      Up
ip4ip4 10    Up      Up
```

```
esr# show ip interfaces gigabitethernet 1/0/16
```

IP address	Type	Precedence	Interface	Admin	Link
16.0.0.2/24			gi1/0/16	Up	Up

```
DHCP --
DHCP Client settings:
DHCP Server: N/A
Lease time(d:h:m): 000:02:00
Reboot time: 10 seconds
Retry time: 300 seconds
Timeout: 60 seconds
Select timeout: 0 seconds
Vendor class ID: N/A
Ignore options:
router
```

```
Latest lease contents:
Lease time(d:h:m): 000:02:00
DHCP message type: DHCPACK
Renew at: 2015-02-25 12:22:24
Rebind at: 2015-02-25 13:14:09
Expires at: 2015-02-25 13:29:09
```

## 19 Настройка IPv6-адресации

- [ipv6 address](#)
- [ipv6 enable](#)
- [ipv6 nd managed-config-flag](#)
- [ipv6 nd ns-interval](#)
- [ipv6 nd other-config-flag](#)
- [ipv6 nd prefix](#)
- [ipv6 nd ra hop-limit](#)
- [ipv6 nd ra lifetime](#)
- [ipv6 nd ra max-interval](#)
- [ipv6 nd ra min-interval](#)
- [ipv6 nd router-preference](#)
- [ipv6 redirects](#)
- [ipv6 route source-route](#)
- [ipv6 unreachable](#)
- [show ipv6 interfaces](#)

### ipv6 address

Данной командой создаётся IPv6-интерфейс и добавляется IPv6-адрес и маска подсети для конфигурируемого интерфейса: физического интерфейса, группы агрегации каналов, туннеля или сетевого моста.

Использование отрицательной формы команды (no) удаляет IPv6-адрес с интерфейса. При удалении последнего адреса IPv6-интерфейс уничтожается.

**!** При создании IPv6-интерфейса система резервирует наибольший незанятый VLAN ID, который будет использоваться внутри системы. Для каждого IPv6-интерфейса на Ethernet-порту резервируется VLAN. Если VLAN уже был зарезервирован для IP-интерфейса, то для IPv6-интерфейса VLAN резервироваться не будет. Можно зарезервировать VLAN ID для внутреннего использования явно с помощью команды `ip internal-usage vlan <VLAN_ID>`.

### Синтаксис

```
ipv6 address <IPV6-ADDR/LEN> [ unit <ID> ]
no ipv6 address {<IPV6-ADDR/LEN>|all} [ unit <ID> ]
```

### Параметры

<IPV6-ADDR/LEN> – IP-адрес и префикс подсети, задаётся в виде X:X:X:X::X/EE, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128]. Можно указать несколько IPv6-адресов перечислением через запятую. Может быть назначено до 8 IPv6-адресов (включая IP-адреса) на интерфейс. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все IPv6-адреса;

<ID> – номер юнита, принимает значения [1..2].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-OOB  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-LOOPBACK  
CONFIG-LT

### Пример

```
esr(config-if-gi)# ipv6 address fc00::1/120
```

### ipv6 enable

Данной командой включается поддержка IPv6 на интерфейсе.

Использование отрицательной формы команды (no) отключает поддержку IPv6 на интерфейсе.

### Синтаксис

```
[no] ipv6 enable
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Поддержка IPv6 отключена.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-OOB  
CONFIG-IF-SUB

CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-LOOPBACK  
CONFIG-LT

### Пример

```
esr(config-if-gi)# ipv6 enable
```

### ipv6 nd managed-config-flag

Данной командой устанавливается managed-config-flag в SLAAC-сообщениях. Использование данного флага определяет получение всех IPv6-настроек от DHCPv6-сервера.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
[no] ipv6 nd managed-config-flag
```

### Параметры

Отсутствуют.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-LOOPBACK  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE

## Пример

```
esr(config)# ipv6 nd managed-config-flag
```

## ipv6 nd ns-interval

Данной командой устанавливается интервала отправки IPv6-сообщений поиска соседа (neighbor solicitation) для широковещательного домена.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
ipv6 nd ns-interval <TIME>
```

```
no ipv6 nd ns-interval
```

## Параметры

<TIME> – интервал отправки, определяется в миллисекундах [1..172800000].

## Значение по умолчанию

0

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-LOOPBACK

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

## Пример

```
esr(config)# ipv6 nd ns-interval 2000
```

## ipv6 nd other-config-flag

Данной командой устанавливается other-config-flag в SLAAC-сообщениях. Использование данного флага подразумевает предоставление IPv6-префикса по протоколу SLAAC, а остальных параметров — по протоколу DHCPv6.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
[no] ipv6 nd other-config-flag
```

### Параметры

Отсутствуют.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-LOOPBACK

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

### Пример

```
esr(config)# ipv6 nd other-config-flag
```

## ipv6 nd prefix

Данной командой устанавливается IPv6-префикс для рассылаемых SLAAC-сообщений для широковещательного домена.

Использование отрицательной формы команды (no).

**Синтаксис**

```
ipv6 nd prefix <IPV6-ADDR/LEN> <VLAID-TIME> <PREFER-TIME> [no-autoconfig] [no-onlink]
no ipv6 nd prefix
```

**Параметры**

<IPV6-ADDR/LEN> – IP-адрес и префикс подсети, задаётся в виде X:X:X:X::X/EE, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128];

<VLAID-TIME> – допустимое время жизни анонса, определяется в секундах [1..2147483647];

<PREFER-TIME> – предпочтительное время жизни анонса, определяется в секундах [1..2147483647];

[no-autoconfig] – не использовать Neighbor Discovery для установки всех действительных префиксов на канале из объявлений маршрутизатора (RA), полученных на интерфейсе;

[no-onlink] – конфигурирует указанный префикс как не on-link. Префикс будет объявлен с установленным L-битом.

**Значение по умолчанию**

Префикс не установлен;

valid time – 0;

preffered time – 0;

no-onlink – не установлен;

no-autoconfig – не установлен.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-LOOPBACK

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

**Пример**

```
esr(config)# ipv6 nd prefix
```



## ipv6 nd ra hop-limit

Данной командой устанавливается параметр hop-limit для рассылаемых SLAAC-сообщений для широковещательного домена.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ipv6 nd ra hop-limit <HOP-LIMIT>
```

```
no ipv6 nd ra hop-limit
```

### Параметры

<HOP-LIMIT> – задается в диапазоне [0..255].

### Значение по умолчанию

64

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-LOOPBACK

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

### Пример

```
esr(config)# ipv6 nd ra hop-limit 128
```

## ipv6 nd ra lifetime

Данной командой устанавливается параметр lifetime для рассылаемых SLAAC-сообщений для широковещательного домена.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

### Синтаксис

```
ipv6 nd ra lifetime <TIME>  
no ipv6 nd ra lifetime
```

### Параметры

<TIME> – время жизни рассылаемых объявлений маршрутизатора, определяется в секундах [0..9000].

### Значение по умолчанию

Не ограничено.

### Необходимый уровень привилегий

10

### Командный режим

```
CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-LOOPBACK  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE
```

### Пример

```
esr(config)# ipv6 nd ra lifetime 120
```

### ipv6 nd ra max-interval

Данной командой устанавливается максимальный интервал для рассылаемых SLAAC-сообщений для широковещательного домена.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ipv6 nd ra max-interval <TIME>  
no ipv6 nd ra max-interval
```

### Параметры

<TIME> – максимальный интервал рассылки SLAAC-сообщений, определяется в секундах [4..1800].

**Значение по умолчанию**

600

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-LOOPBACK

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

**Пример**

```
esr(config)# ipv6 nd ra max-interval 1200
```

**ipv6 nd ra min-interval**

Данной командой устанавливается минимальный интервал для рассылаемых SLAAC-сообщений для широковещательного домена.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
ipv6 nd ra min-interval <TIME>
```

```
no ipv6 nd ra min-interval
```

**Параметры**

<TIME> – минимальный интервал рассылки SLAAC-сообщений, определяется в секундах [3..1350].

**Значение по умолчанию**

200

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-LOOPBACK

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

**Пример**

```
esr(config)# ipv6 nd ra min-interval 30
```

**ipv6 nd router-preference**

Данной командой устанавливается приоритетность маршрутизатора в процессе выбора маршрута по умолчанию для широковещательного домена.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
ipv6 nd router-preference <ACTION>
```

```
no ipv6 nd router-preference
```

**Параметры**

<ACTION> – принимает значения:

- low;
- medium;
- high.

**Значение по умолчанию**

medium

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-LOOPBACK  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE

### Пример

```
esr(config)# ipv6 nd router-preference low
```

### ipv6 redirects

Данной командой включается механизм отправки ICMP-сообщений о существовании более приоритетного маршрутизатора в данной IPv6-сети для конкретного IPv6-назначения.

Использование отрицательной формы команды (no) отключает механизм отправки ICMP-сообщений о существовании более приоритетного маршрутизатора в данной IPv6-сети для конкретного IPv6-назначения.

### Синтаксис

```
[no] ipv6 redirects
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

15

### Значение по умолчанию

Отправка ICMP-сообщений о существовании более приоритетного маршрутизатора в данной IP-сети для конкретного IP-назначения включена.

### Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-OOB  
CONFIG-IF-SUB

CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-LOOPBACK

### Пример

```
esr(config-if-gi)# no ipv6 redirects
```

### ipv6 route source-route

Данной командой на маршрутизаторе включается поддержка опции ipv6 source-route.

Использование отрицательной формы команды (no) отключает поддержку опции ipv6 source-route.

### Синтаксис

```
[no] ipv6 route source-route
```

### Параметры

Отсутствуют.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# ipv6 route source-route
```

### ipv6 unreachablees

Данной командой включается отправка IPv6 ICMP-пакетов о недоступности конечного адреса.

Использование отрицательной формы команды (no) отключает возможность отправки IPv6 ICMP-пакетов о недоступности конечного адреса.

### Синтаксис

```
[no] ipv6 unreachablees
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Отправка IPv6 ICMP-пакетов о недоступности конечного адреса включена.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

## Пример

```
esr(config-if-gi)# no ipv6 unreachable
```

## show ipv6 interfaces

Команда используется для просмотра информации о существующих в системе IPv6-интерфейсах.

## Синтаксис

```
show ipv6 interfaces { { <IF> | <TUN> | ipv6-address <IPV6-ADDR> } | vrf <VRF> [ ipv6-address <IPV6-ADDR> ] }
```

## Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа. При указании данного параметра будут отображены IPv6-интерфейсы в указанном VRF;

<IF> – наименования системных интерфейсов, задаются в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<TUN> – наименования туннелей, задаются в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#);

<IPv6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X/EE, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128]. При указании данного параметра будет отображен IPv6-интерфейс с указанным IPv6-адресом.

В команде можно указать несколько системных интерфейсов. Если не указывать индексы интерфейсов, то будут отображены все IPv6-интерфейсы, относящиеся к системным интерфейсам указанного типа.

Если в команде указан определенный системный интерфейс, получающий IPv6-параметры по протоколу DHCP, то будут отображены настройки DHCP-клиента и состояние текущей аренды IPv6-параметров.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```

esr# show ipv6 interfaces
IPv6 address          Interface          Admin  Link
Type
-----
-----
fc00::2/120          gi1/0/5           Up     Up
DHCP
fe80::aaf9:4bff:feaa:45/64
static

esr# show ip interfaces gigabitethernet 1/0/16
IPv6 address          Interface          Admin  Link
Type
-----
-----
fc00::2/120          gi1/0/16          Up     Up
DHCP
  DHCP Client settings:
  DHCP Server:         N/A
  Lease time(d:h:m):   000:02:00
  Reboot time:         10 seconds
  Retry time:          300 seconds
  Timeout:              60 seconds
  Select timeout:      0 seconds
  Vendor class ID:     N/A
  Ignore options:
    router
  Latest lease contents:
  Lease time(d:h:m):   000:02:00
  DHCP message type:   DHCPACK
  Renew at:            2015-02-25 12:22:24
  Rebind at:           2015-02-25 13:14:09
  Expires at:          2015-02-25 13:29:09

```



## 20 Управление профилями

- address-port pair
- ap-model
- application
- category
- description
- email
- ip address-range
- ip prefix
- ipv6 address-range
- ipv6 prefix
- mac address
- object-group address-port
- object-group ap-models
- object-group application
- object-group content-filter
- object-group mac
- object-group network
- object-group service
- object-group url
- port-range
- regexp
- show object-group
- url
- vendor

### address-port pair

Команда используется для задания связки IP-адресов и TCP/UDP-порта.

Использование отрицательной формы команды (no) удаляет запись из конфигурируемого профиля.

#### Синтаксис

```
[no] address-port pair { <IPv4-ADDR> | <IPv6-ADDR> } : <PORT>
```

#### Параметры

<IPv4-ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPv6-ADDR> – IPv6-адрес, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<PORT> – номер порта, принимает значение [1..65535].

#### Необходимый уровень привилегий

10


#### Командный режим

CONFIG-OBJECT-GROUP-ADDRESS-PORT

## Пример

```
esr(config-object-group-address-port)# address-port pair 192.168.1.1:23
```

## ap-model

 Данная команда описана в разделе [Управление профилями ap-models](#).

## application

Данная команда используется для указания приложений, попадающих под действие данного профиля. Использование отрицательной формы команды (no) удаляет приложение из текущего профиля.

## Синтаксис

```
[no] application < APPLICATION >
```

## Параметры

< APPLICATION > – указывает приложение, подпадающее под действие данного профиля, может принимать значения:

- afp – Apple Filing Protocol;
- amazon – Amazon Data Services;
- amazon-video - Amazon Video Services;
- amqp – Advanced Message Queuing Protocol;
- apple – Apple Inc.;
- apple-icloud – Apple iCloud;
- apple-itunes – Apple iTunes;
- apple-push – Apple PUSH notifications;
- apple-store – Apple Store;
- applejuice – Applejuice P2P;
- ajp – Apache JServ Protocol;
- signal – Signal Protocol;
- ayiya – Anything In Anything;
- battlefield – Battlefield;
- bgp – Border Gateway Protocol;
- bittorrent – BitTorrent;
- bjnp – Canon BJNP protocol;
- cisco-skinny – Cisco Skinny;
- cisco-vpn – Cisco VPN;
- citrix – Citrix;
- cloudflare – Cloudflare Inc.;
- soap – Constrained Application Protocol;
- collectd – Collectd;
- corba – Common Object Request Broker Architecture;
- dce-rpc – Distributed Computing Environment / Remote Procedure Calls;
- deezer – Deezer (music streaming service).;
- dhcp – Dynamic Host Configuration Protocol;
- dhcpv6 – IPv6 Dynamic Host Configuration Protocol;
- directconnect – Direct Connect;
- dns – Domain Name System;
- dnscrypt – DNSCrypt;

- drda – Distributed Relational Database Architecture;
- dropbox – Dropbox;
- ebay – eBay;
- edonkey – eDonkey;
- egp – Exterior Gateway Protocol;
- facebook – Facebook;
- facebook-zero – Facebook Zero;
- fasttrack – FastTrack P2P;
- ftp-control – File Transfer Protocol (control connections);
- ftp-data – File Transfer Protocol (data connections);
- git – Git (version control system);
- github – GitHub, Inc.;
- gmail – Gmail;
- gnutella – Gnutella P2P;
- google – Google Inc.;
- google-maps – Google Maps;
- google-drive – Google Drive;
- google-docs – Google Docs;
- google-plus – Google Plus;
- google-services – Google Services;
- gre – Generic Routing Encapsulation;
- gtp – GPRS Tunneling Protocol;
- h323 – H323 Protocol;
- half-life2 – Half-Life 2;
- imo – IMO Messenger;
- hangout-duo – Google Hangout and Duo Messengers;
- hotmail – Hotmail;
- hotspotshield – Hotspot Shield;
- http – HTTP;
- http-connect – HTTP tunnel;
- http-download – HTTP download data;
- http-proxy – HTTP proxy;
- https – HTTPS;
- iax – IAX;
- icecast – Icecast;
- icmp – ICMP;
- icmpv6 – ICMP IPv6;
- iflix – iflix services;
- igmp – IGMP;
- imap – IMAP;
- imaps – IMAP over TLS/SSL;
- instagram – Instagram;
- ip-in-ip – IP in IP;
- ipp – Internet Printing Protocol;
- ipsec – IPsec;
- irc – Internet Relay Chat;
- kakaotalk – KakaoTalk;
- kakaotalk-voice – KakaoTalk voice;
- kaspersky-lab – Kaspersky Laboratory;
- kaspersky-update – Kaspersky Anti-Virus Update;
- kerberos – Kerberos;
- lastfm – LastFM;
- ldap – Lightweight Directory Access Protocol;
- llmnr – Link-local Multicast Name Resolution;
- linkedin – LinkedIn;
- lotusnotes – Lotus Notes;
- mdns – Multicast DNS;

- messenger – Messenger;
- megaco – Megaco;
- mgcp – Media Gateway Control Protocol;
- microsoft – Microsoft services;
- mining – Mining protocols;
- mpeg-ts – MPEG-TS;
- mqtt – Message Queue Telemetry Transport;
- msn – MSN;
- ms-netlogon – Microsoft Netlogon;
- ms-onedrive – Microsoft OneDrive;
- ms-sms – Microsoft System Management Server;
- mssql-tds – Tabular Data Stream;
- mysql – MySQL;
- netbios – NetBIOS;
- netflix – Netflix;
- netflow – Netflow;
- nest-log-sink – Nest Log Sink Protocol;
- nfs – Network File System;
- noe – New Office Environment protocol;
- ntp – Network Time Protocol;
- office365 – Office365;
- ookla – Ookla services;
- .opendns – OpenDNS;
- openft – OpenFT P2P;
- openvpn – OpenVPN;
- oracle – Oracle Database;
- oscar – OSCAR;
- ospf – Open Shortest Path First;
- pando-media-booster – Pando Media Booster;
- pandora – Pandora Radio;
- pcanywhere – PcAnywhere;
- playstore – Google Playstore;
- pop – Post Office Protocol;
- pops – Post Office Protocol over TLS/SSL;
- postgresql – PostgreSQL;
- pptp – Point-to-Point Tunneling Protocol;
- quic – QUIC;
- radius – Remote Authentication in Dial-In User Service;
- rdp – Remote Desktop Protocol;
- redis – Remote dictionary server;
- rsync – Remote Synchronization;
- rtcp – Real-Time Transport Control Protocol;
- rtmp – Real Time Messaging Protocol;
- rtp – Real-time Transport Protocol;
- rtp-audio – Real-time Transport Protocol audio content;
- rtp-dynamic – Real-time Transport Protocol dynamic content;
- rtp-video – Real-time Transport Protocol video content;
- rtsp – Real time streaming protocol;
- sctp – Stream Control Transmission Protocol;
- sflow – sFlow;
- shoutcast – SHOUTcast;
- sip – Session Initiation Protocol;
- skype – Skype;
- smb – Server Message Block;
- smbv23 – Server Message Block Version 23;
- smpp – Short Message Peer-to-Peer;
- smtp – Simple Mail Transfer Protocol;

- smtps – Simple Mail Transfer Protocol over TLS/SSL;
- snmp – Simple Network Management Protocol;
- socks – SOCKS Protocol;
- soulseek – Soulseek P2P;
- soundcloud – Soundcloud;
- ssdp – Simple Service Discovery Protocol;
- ssh – Secured Shell;
- starcraft – StarCraft;
- stealthnet – StealthNet P2P;
- steam – Steam;
- stun – Session Traversal Utilities for NAT;
- syslog – System Logging Utility;category
- teamspeak – TeamSpeak;
- teamviewer – TeamViewer;
- telegram – Telegram;
- telnet – Teletype network;
- teredo – Teredo tunneling;
- tftp – Trivial File Transfer Protocol;
- tor – The Onion Router;
- tls – Transport Layer Security Protocol;
- twitch – Twitch;
- twitter – Twitter;
- unencrypted-jabber – Unencrypted Jabber;
- viber – Viber;
- vmware-vmware – VMWare vSphere;
- vnc – Virtual Network Computing;
- vrrp – Virtual Router Redundancy Protocol;
- warcraft3 – Warcraft 3;
- wechat – WeChat;
- whatsapp – WhatsApp;
- whatsapp-call – WhatsApp call;
- whatsapp-files – WhatsApp file;
- whois-das – Whois/DAS;
- wikipedia – Wikipedia;
- windows-update – Windows update services;
- world-of-warcraft – World of Warcraft;
- xbox – Xbox services;
- xdmcp – X Display Manager control protocol);
- yahoo – Yahoo services;
- youtube – YouTube;
- zabbix – Zabbix;
- zeromq – ZeroMQ;
- zoom – Zoom call.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-OBJECT-GROUP-APPLICATION

## Пример

```
esr(config-object-group-application)# application egp
```

## category

Команда используется для задания категорий контентной фильтрации.

Использование отрицательной формы команды (no) удаляет заданную категорию.

### Синтаксис

```
[no] category <CATEGORY>
```

### Параметры

<CATEGORY> – имя категории. Принимает значения:

- `addictive-substances` – мастер-категория, включающая контент, связанный с алкоголем, табаком, электронными сигаретами и психотропными веществами.
- `addictive-substances alcohol` – контент, связанный с алкогольной продукцией, производством, потреблением, рекламой, продажей, сайтами производителей.
- `addictive-substances narcotics` – контент, связанный с наркотическими, психотропными и одурманивающими веществами, инструкцией по применению, действием, сопровождением немедицинской зависимости.
- `addictive-substances tobacco` – контент, связанный с табачными изделиями, электронными сигаретами, описанием использования, рекламой, продажей.
- `adware` – контент, связанный с рекламным или потенциально нежелательным программным обеспечением (PUA).
- `blocked-by-legistations` – мастер-категория, включающая контент, заблокированный законодательством отдельных стран.
- `blocked-by-legistations bgc-list` – контент, заблокированный Бельгийской комиссией по азартным играм.
- `blocked-by-legistations japanese-police` – контент, запрещенный Японской полицией.
- `blocked-by-legistations sia-japan` – пиратские сайты, запрещенные Ассоциацией безопасного Интернета (SIA) Японии.
- `downloadable-content` – мастер-категория, включающая контент связанный с загрузкой пакетов программного обеспечения.
- `downloadable-content file-sharing` – контент, связанный с хранением, синхронизацией и обменом файлами пользователя.
- `downloadable-content media-content` – контент, связанный с распространением аудио и видео материалов, фильмов, спортивных трансляций, концертов, песен, клипов, обучающих программ.
- `downloadable-content torrents` – контент, связанный с Торрент-ресурсами для однорангового обмена файлами.
- `dyn-dns` – служба динамического DNS.
- `eating-disorders` – контент, связанный с расстройствами пищевого поведения, пропагандой или одобрением такого поведения.
- `education` – мастер-категория, включающая контент связанный с веб-сайтами с образованием.
- `education books` – контент, связанный с писателями, книгами, дискуссиями, фан-клубами, издателями, онлайн-библиотеками, книжными магазинами.
- `education educational-institutions` – контент, связанный с школами, университетами, образовательными организациями, онлайн-классами, образовательными веб-сервисами.
- `education encyclopedias` – контент, связанный с сервисами онлайн-обучения, образовательными порталами, энциклопедиями.
- `gambling` – мастер-категория, включающая контент связанный с азартными играми, лотереями, информацией, поощряющей участие в этих мероприятиях.
- `gambling casino` – контент, связанный с казино, азартными карточными играми, инструкцией по игре, советами, информацией об оборудовании казино.
- `gambling online-betting` – контент, связанный с букмекерскими конторами, ставками на спортивные события, скачками, другими событиями.
- `gambling online-lotteries` – контент, связанный с онлайн лотереями, информацией о лотереях.

- hate-speech – мастер-категория, включающая контент связанный с ненавистью, дискриминацией, расизмом, экстремизмом, членовредительством, самоубийством.
- hate-speech extremism – контент, связанный с экстремизмом, терроризмом, расизмом, дискриминацией, ненавистью.
- hate-speech profane-language – контент, связанный с нецензурной речью.
- hate-speech self-harm – контент, связанный с угрозами жизни, членовредительствами или самоубийством.
- hate-speech violence – контент, связанный с оскорблениями, насилием, жестокостью и тревожным контентом с участием людей, животных и экстремистских организаций.
- hobbies-recreation – мастер-категория, включающая контент связанный с хобби, отдыхом, путешествиями, музыкой, домашними животными, развлечениями, кино, театром, телевидением, шоу-бизнесом, фан-клубами, новостями о знаменитостях, сплетнями, комиксами.
- hobbies-recreation animals – контент, связанный с животными, сообществами, организациями, инструкцией по уходу за животными, онлайн-трансляции.
- hobbies-recreation cultural-heritage – контент, связанный с культурным наследием, изобразительным искусством, исполнительским искусством, музеями, галереями, выставками.
- hobbies-recreation games – контент, связанный с информация о видеоиграх, обсуждениях, онлайн-играх, браузерных играх, поддержке игрового программного обеспечения.
- hobbies-recreation humour – контент, связанный с юмором, анекдотами, мемами, юмористическими историями, смешными видео, комедийными шоу.
- hobbies-recreation hunting-fishing – контент, связанный с охотой и рыбалкой, сообществами, советами, магазинами снаряжения.
- hobbies-recreation motor-vehicles – контент, связанный с информацией об автомобилях, обзорах, покупках, модификациях, ремонте, общественном транспорте.
- hobbies-recreation music – контент, связанный с музыкой, текстами песен, нотами, официальными страницами музыкантов, билетами на концерты, музыкальными заведениями, стриминговыми платформами, музыкальными теле/радиоканалы.
- hobbies-recreation traveling – контент, связанный с путешествиями, гидами, бронированием отелей и билетов, обзорами, туристическими агентствами, обсуждениями туризма.
- hobbies-recreation tv-radio – контент, связанный с телекомпаниями, онлайн ТВ, радиокомпаниями, онлайн радиовещателями.
- human-life – мастер-категория, включающая контент связанный с религией, политикой, законами, домом и семьей, средствами массовой информации, армией, оружием, поиском работы, едой, астрологией и паранормальными явлениями.
- human-life business – контент, связанный с предприятиями, отраслевыми группами и корпоративными услугами.
- human-life family – контент, связанный с семейным образом жизни, ведением домашнего хозяйства, проектами DIY, ремонтом дома, украшением дома.
- human-life finance – контент, связанный с финансами, экономикой, сопутствующими услугами.
- human-life food – контент, связанный с ресторанами, кафе, рецептами, технологиями приготовления еды, кулинарными сообществами, продуктовыми магазинами.
- human-life government – контент, связанный с правоохранительными органами, государственными учреждениями, политическими объединениями, адвокатскими конторами, юридическими консультациями.
- human-life magic – контент, связанный с мистикой и эзотерическими темами, гороскопами, гаданиями, магией, НЛО, паранормальными явлениями.
- human-life military – контент, связанный с военными, вооруженными силами, подрядчиками, доктринами, дискуссиями по вооруженным конфликтам.
- human-life news – контент, связанный с публичным новостным контентом, официальными СМИ, информационными службами, агрегаторами новостей, новостными сайтами, создаваемыми пользователями.
- human-life recruitment – контент, связанный с объединениями работодателей и соискателями работы.
- human-life religion – контент, связанный с религиозными движениями, организациями, культурами и межконфессиональным сотрудничеством.
- human-life weapons – контент, связанный с оружием, охотой, выживанием, стрелковым спортом, военной тематикой.

- it-services – мастер-категория, включающая контент связанный с ИТ-услугами и технологическим оборудованием, анонимайзерами, поисковыми системами, хостинг, домены, рекламные сети, интернет-сервисы, компьютеры, электроника, информационная безопасность, спам-сайты.
- it-services ads – контент, связанный с рекламными агентствами, рекламными сетями.
- it-services anonymizers – контент, связанный с анонимным доступом и обходом ограничений, в том числе VPN.
- it-services electronics – контент, связанный с информацией о технической и бытовой электронике, новостями, инструкциями, обзорами, сайтами производителей.
- it-services hosting – контент, связанный с хостингом, регистрацией доменов, услугами поиска WHOIS.
- it-services internet-services – контент, связанный с поставщиками интернет-инфраструктуры и услуг, Интернет-провайдеров, онлайн-переводчиками, аналитикой веб-сайтов, хостингом для разработки программного обеспечения, веб-сервисами.
- it-services it-security – контент, связанный с ИТ-безопасностью и продуктами/услугами по обеспечению безопасности.
- it-services searchers – контент, связанный с базой данных интернет-ресурсов со встроенной поисковой системой.
- it-services spam – контент, связанный с рассылкой спама или использованием в спаме.
- kids-internet – контент, связанный с сайтами, ориентированными на детей, посвященные образовательным, развлекательным и коммуникационным платформам.
- lifestyle – мастер-категория, включающая контент связанный со спортом, красотой и здоровьем.
- lifestyle fashion – контент, связанный с модой и стилем, модельные агентства, показы мод, гиды по одежде и обуви.
- lifestyle health – контент, связанный со здоровым образом жизни, фитнесом, здоровым питанием, альтернативными методами лечения.
- lifestyle pharmacy – контент, связанный с медициной, аптеками, медицинскими организациями, реабилитационными центрами, фармацевтическими компаниями, лекарствами, БАД, профилактикой инфекций, медицинской психологической помощью.
- lifestyle sport – контент, связанный со спортом, историей спорта, правилами игры, сообществами, командами, легкой атлетикой, болельщиками, новостями спорта.
- malware – контент, связанный с вредоносными ресурсами и URL-адресами загрузки программного обеспечения.
- payments – мастер-категория, включающая контент связанный с Интернет-магазинами, платежными системами, банками, онлайн-кошельками и криптовалютой.
- payments cryptocurrency – контент, связанный с покупкой, продажей, хранением и изучением криптовалют и криптомайнинга.
- payments online-banks – контент, связанный с услугами онлайн-банкинга и электронного кошелька.
- payments shops – контент, связанный с Интернет-магазинами и аукционами по продаже товаров или услуг.
- payments transactions – контент, связанный с системами электронных денег, доступом к личным кабинетам и возможность проведения платежей.
- rat-node – контент, связанный с программным обеспечением инструментов удаленного администрирования.
- rental-services – контент, связанный с арендой, продажей и покупкой недвижимости.
- rf-law-prohibited – мастер-категория, включающая контент, заблокированный законодательством Российской Федерации.
- rf-law-prohibited 436-fz – категория соблюдения законодательства, веб-ресурсы, предусмотренные статьей 5 Федерального закона 436-ФЗ.
- rf-law-prohibited extremist-rf-minjust – контент, внесенный в федеральный реестр экстремистских материалов, который ведется Министерством юстиции Российской Федерации.
- rf-law-prohibited roskomnadzor – контент, включенный в единый реестр запрещенных Интернет-ресурсов Роскомнадзора.
- scam – контент, связанный с угрозами потери данных, фишинговыми атаками.
- sex – мастер-категория, включающая контент, связанный с порнографией, эротикой, сексуальностью, нудизмом, половым воспитанием, сексуальными услугами онлайн или оффлайн, продуктами и услугами сексуального характера, информацией об абортax.



- **sex erotic-18+** – контент откровенно сексуального характера, предназначенный для взрослой аудитории, часто представленный в возбуждающей манере.
- **sex lgbtq+** – контент, связанный с LGBTQ+ темами.
- **sex lingerie** – контент, связанный с производителями, продавцами нижнего белья и модными страницами с описаниями или фотографиями нижнего белья.
- **sex near-sexual-content** – контент, связанный с информацией о сексуальности, сексуальном здоровье, контрацепции и смежных темах, подходящей для взрослой аудитории, но не предназначенной для детей.
- **social-media** – мастер-категория, включающая контент, связанный с отправкой личных сообщений или публикаций общедоступных сообщений или сообщений для ограниченной аудитории.
- **social-media blogs** – контент, связанный с блог-платформами, персональными блогами, сервисами для создания и ведения блогов.
- **social-media chats** – контент, связанный с публичными обсуждениями и обменом мгновенными сообщениями в режиме реального времени.
- **social-media dating** – контент, связанный с услугами онлайн и оффлайн знакомств.
- **social-media email** – контент, связанный со страницами входа в почтовые сервисы и почтовые ящики.
- **social-media social-networks** – контент, связанный с созданием, отображением и управлением контактами между людьми, организациями и правительствами.
- **tor** – контент, связанный с узлами Tor.
- **uncategorized** – некатегоризированный контент.
- **vulnerables** – контент, связанный с уязвимым программным обеспечением, которым могут воспользоваться злоумышленники.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-OBJECT-GROUP-CF-KASPERSKY

### Пример

```
esr(config-object-group-cf-kaspersky)# category social-media chats
```

### description

Команда используется для изменения описания профиля.

Использование отрицательной формы команды (**no**) удаляет описание профиля.

### Синтаксис

```
description <DESCRIPTION>
```

```
no description
```

### Параметры

<DESCRIPTION> – описание профиля, задаётся строкой до 255 символов.

### Необходимый уровень привилегий

10

**Командный режим**

CONFIG-OBJECT-GROUP-NETWORK  
 CONFIG-OBJECT-GROUP-SERVICE  
 CONFIG-OBJECT-GROUP-MAC  
 CONFIG-OBJECT-GROUP-APPLICATION  
 CONFIG-OBJECT-GROUP-URL  
 CONFIG-OBJECT-GROUP-ADDRESS-PORT  
 CONFIG-OBJECT-GROUP-CONTENT-FILTER  
 CONFIG-OBJECT-GROUP-MAIL

**Пример**

Установить описание для профиля IP-адресов:

```
esr(config-object-group-network)# description "Internal addresses"
```

**email**

Команда используется для задания почтового домена или адреса почтового ящика.

Использование отрицательной формы команды (no) удаляет запись из конфигурируемого профиля.

**Синтаксис**

```
[no] email <NAME>
```

**Параметры**

<NAME> – почтовый домен или адреса почтового ящика, задаётся строкой от 1 до 31 символа.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-OBJECT-GROUP-MAIL

**Пример**

```
esr(config-object-group-email)# email eltex@eltex-co.ru
esr(config-object-group-email)# email eltex-co.ru
```

**ip address-range**

Команда используется для задания диапазона IP-адресов.

Использование отрицательной формы команды (no) удаляет запись из конфигурируемого профиля.

**Синтаксис**

```
[no] ip address-range <FROM-ADDR>[-<TO-ADDR>] [ unit <ID> ]
```

**Параметры**

<FROM-ADDR> – начальный IP-адрес диапазона адресов;

<TO-ADDR> – конечный IP-адрес диапазона адресов, опциональный параметр. Если параметр не указан, то командой задаётся одиночный IP-адрес;

Адреса задаются в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<ID> – номер юнита, принимает значения [1..2];

Для ESR-20/21/30/31/100/200/1000/1200/1500/1700 возможно указать до 64 отдельных IP-диапазонов в рамках одной группы адресов;

Для ESR-10/12V/12VF/15/15R/15VF возможно указать до 6 отдельных IP-диапазонов в рамках одной группы адресов.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-OBJECT-GROUP-NETWORK

**Пример**

```
esr(config-object-group-network)# ip address-range 192.168.1.1-192.168.1.25
```

**ip prefix**

Команда используется для задания подсети.

Использование отрицательной формы команды (no) удаляет заданную подсеть.

**Синтаксис**

```
[no] ip prefix <ADDR/LEN> [ unit <ID> ]
```

**Параметры**

<ADDR/LEN> – IP-подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [8..32];

<ID> – номер юнита, принимает значения [1..2].

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-OBJECT-GROUP-NETWORK

## Пример

```
esr(config-object-group-network)# ip prefix 10.10.10.0/24
```

## ipv6 address-range

Команда используется для задания диапазона IPv6-адресов. Использование отрицательной формы команды (no) удаляет запись из конфигурируемого профиля.

### Синтаксис

```
[no] ipv6 address-range <FROM-ADDR>[-<TO-ADDR>] [ unit <ID> ]
```

### Параметры

<FROM-ADDR> – начальный IPv6-адрес диапазона адресов;

<TO-ADDR> – конечный IPv6-адрес диапазона адресов, опциональный параметр. Если параметр не указан, то командой задаётся одиночный IPv6-адрес;

Адреса задаются в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<ID> – номер юнита, принимает значения [1..2];

Для ESR-20/21/30/31/100/200/1000/1200/1500/1700 возможно указать до 64 отдельных IP-диапазонов в рамках одной группы адресов;

Для ESR-10/12V/12VF/15/15R/15VF возможно указать до 6 отдельных IP-диапазонов в рамках одной группы адресов.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-OBJECT-GROUP-NETWORK

## Пример

```
esr(config-object-group-network)# ipv6 address-range fc00::1:1-fc00:1::32
```

## ipv6 prefix

Команда используется для задания IPv6-подсети.

Использование отрицательной формы команды (no) удаляет заданную подсеть.

### Синтаксис

```
[no] ipv6 prefix <IPV6-ADDR/LEN> [ unit <ID> ]
```

## Параметры

<IPv6-ADDR/LEN> – IP-адрес и маска подсети, задаётся в виде X:X:X::X/EE, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128];

<ID> – номер юнита, принимает значения [1..2].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-OBJECT-GROUP-NETWORK

## Пример

```
esr(config-object-group-network)# ipv6 prefix fc00::/126
```

## mac address

Данная команда используется для задания диапазона MAC-адресов.

Использование отрицательной формы команды (no) удаляет запись из конфигурируемого профиля.

## Синтаксис

```
[no] mac address <MAC-ADDR> <MAC-MASK>
```

## Параметры

<MAC-ADDR> – MAC-адрес, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF];

<MAC-MASK> – маска MAC-адреса, опциональный параметр, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF]. Биты маски, установленные в 0, задают биты MAC-адреса, исключаемые из сравнения при поиске. Значение маски по умолчанию FF:FF:FF:FF:FF:FF.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-OBJECT-GROUP-MAC

## Пример

```
esr(config-object-group-mac)# mac address a8:f9:4b:80:e7:00 FF:FF:FF:FF:FF:00
```

## object-group address-port

Команда предназначена для создания профиля связок IP-адресов и TCP/UDP-портов. Профили используются при настройке сервисов, работающих с пулами IP-адресов и TCP/UDP-портами – NAT, Firewall.

Использование отрицательной формы команды (no) удаляет профиль IP-адресов.

### Синтаксис

```
[no] object-group address-port <NAME>
```

### Параметры

<NAME> – имя конфигурируемого профиля связок IP-адресов и TCP/UDP-портов, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра "all" будут удалены все профили IP-адресов.

### Необходимый уровень привилегий

10

### Командный режим


CONFIG

### Пример

Создание профиля IP-адресов с именем *remote* и переход в режим конфигурирования профиля:

```
esr(config)# object-group address-port WEB
```

## object-group ap-models

 Данная команда описана в разделе [Управление профилями ap-models](#).

## object-group application

Данная команда используется для создания профиля приложений. Данный профиль используется для фильтрации по приложениям (DPI).

Использование отрицательной формы команды (no) удаляет профиль.

### Синтаксис

```
[no] object-group application <NAME>
```

### Параметры

<NAME> – имя профиля приложений, задается строкой до 31 символа. При удалении вместо имени возможно использовать ключ all. При использовании ключа "all" будут удалены все профили приложений.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# object-group application OGA045
```

**object-group content-filter**

Данная команда используется для создания профиля категорий контентной фильтрации. Данный профиль используется в наборе пользовательских правил.

Использование отрицательной формы команды (no) удаляет профиль.

**Синтаксис**

```
[no] object-group content-filter <NAME>
```

**Параметры**

<NAME> – имя профиля контентной фильтрации, задается строкой до 31 символа. При удалении вместо имени возможно использовать ключ all. При использовании ключа "all" будут удалены все профили контентной фильтрации.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# object-group content-filter OGC042
```

**object-group mac**

Данная команда используется для создания профиля MAC-адресов. Данный профиль используется в MAC-based VLAN.

Использование отрицательной формы команды (no) удаляет профиль.

**Синтаксис**

```
[no] object-group mac <NAME>
```

## Параметры

<NAME> – имя профиля MAC-адресов, задается строкой до 31 символа. При удалении вместо имени возможно использовать ключ all. При использовании ключа "all" будут удалены все профили MAC-адресов.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# object-group mac OGM007
```

## object-group network

Команда предназначена для создания профиля IP-адресов. Профили используются при настройке сервисов, работающих с пулами IP-адресов – например, NAT, Firewall, Remote-Access, также для создания списка префиксов.

Использование отрицательной формы команды (no) удаляет профиль IP-адресов.

## Синтаксис

```
[no] object-group network <NAME>
```

## Параметры

<NAME> – имя конфигурируемого профиля IP-адресов, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все профили IP-адресов.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

Создание профиля IP-адресов с именем *remote* и переход в режим конфигурирования профиля:

```
esr(config)# object-group network remote
```



## object-group service

Команда предназначена для создания профиля TCP/UDP-портов. Данный профиль используется в правилах сервисов NAT и Firewall.

Использование отрицательной формы команды (no) удаляет профиль.

### Синтаксис

```
[no] object-group service <NAME>
```

### Параметры

<NAME> – наименование профиля портов, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все профили TCP/UDP-портов.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# object-group service ssh
```

## object-group url

Команда предназначена для создания профиля URL-ссылок.

Использование отрицательной формы команды (no) удаляет профиль.

### Синтаксис

```
[no] object-group url <NAME>
```

### Параметры

<NAME> – наименование профиля портов, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все профили URL-ссылок.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

## Пример

```
esr(config)# object-group url vk
```

## port-range

Командой задаётся диапазон TCP/UDP-портов, относящихся к профилю.

Использование отрицательной формы команды (no) удаляет запись из конфигурируемого профиля.

## Синтаксис

```
port-range <PORT>
```

```
no port-range [<PORT> | all]
```

## Параметры

<PORT> – номер порта, принимает значение [1..65535].

Можно указать несколько портов перечислением через запятую «,» либо указать диапазон портов через «-». Пример записи: <PORT>, <PORT> или <PORT>-<PORT> или <PORT>-<PORT>, <PORT>-<PORT>.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-OBJECT-GROUP-SERVICE

## Пример

```
esr(config-object-group-service)# port-range 22
```

## regex

Данной командой описывается шаблон URL-ссылок.

Использование отрицательной формы команды (no) удаляет шаблон URL-ссылок.

## Синтаксис

```
regex <REGEXP>
```

```
no regex {<REGEXP>|all}
```

## Параметры

<REGEXP> – регулярное выражение. Описывается строкой до 255 символов. Символ "\" должен быть экранирован.

all – ключ, используемый для удаления всех созданных правил.

**Значение по умолчанию**

Шаблон не создан.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-OBJECT-GROUP-URL

**Пример**

```
esr(config-object-group-url)# '^http://\/site\.ru'
```

**show object-group**

Данная команда используется для просмотра информации о профилях IP-адресов и TCP/UDP-портов.

**Синтаксис**

```
show object-group <PROFILE_TYPE> [<NAME>]
```

**Параметры**

<PROFILE\_TYPE> – тип профиля:

- address-port – профиль связей IP-адресов и TCP/UDP-портов;
- application – профиль приложений;
- content-filter – профиль контентной фильтрации;
- email – профиль почтовых доменов и адресов почтовых ящиков;
- mac – профиль профиля MAC-адресов;
- network – профиль IP-адресов;
- service – профиль TCP/UDP-портов;
- url – профиль URL-ссылок.

<NAME> – имя профиля, задаётся строкой до 31 символа, опциональный параметр. Если имя профиля не задано, то будет выведена информация по всем профилям IP-адресов и TCP/UDP-портов.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример**

```
esr# show object-group network
Network          Description
-----
```

```

remote                --
local                 --
tunnel                --
esr# show object-group network remote
IP Addresses
-----
10.102.0.0/16
esr# show object-group service
Service                Description
-----
telnet                 --
ssh                    --
dhcp_server            --
dhcp_client            --
ntp                    --
esr# show object-group service ssh
Port ranges
-----
22

```

## url

Команда используется для задания URL-ссылки.

Использование отрицательной формы команды (no) удаляет ссылку из конфигурируемого профиля.

### Синтаксис

```

url <URL>
no url [ <URL> | all ]

```

### Параметры

<URL> – текстовое поле, содержащее URL-ссылку длиной от 4 до 255 символов.

При удалении с использованием ключа "all" будут удалены все внесенные ранее URL-ссылки.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-OBJECT-GROUP-URL

### Пример

```

esr(config-object-group-url)# url https://vk.com

```

## vendor

Команда используется для задания поставщика категорий контентной фильтрации.

Использование отрицательной формы команды (no) удаляет категории контентной фильтрации данного поставщика.

## Синтаксис

```
[no] vendor <CONTENT-FILTER-VENDOR>
```

## Параметры

<CONTENT-FILTER-VENDOR> – название поставщика категорий контентной фильтрации. Принимает значения:

Kaspersky-Lab – в текущей версии ПО в качестве поставщика категорий контентной фильтрации может выступать только Лаборатория Касперского.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-OBJECT-GROUP-CONTENT-FILTER

## Пример

```
esr(config-object-group-content-filter)# vendor kaspersky-lab
```

## 21 Управление NAT

- action destination-nat
- action source-nat
- check output-interface
- description
- enable
- from
- ip address
- ip address-range
- ip nat proxy-arp
- ip port
- ip port-range
- match destination-address
- match destination-address-port
- match destination-port
- match icmp
- match protocol
- match source-address
- match source-address-port
- match source-port
- nat alg
- nat destination
- nat source
- persistent
- pool
- rearrange
- renumber rule
- rule
- ruleset
- show ip nat alg
- show ip nat pool
- show ip nat ruleset
- show ip nat translations
- show ip nat proxy-arp
- to

### action destination-nat

Данной командой выполняется трансляция адреса и порта получателя для трафика, удовлетворяющего заданным критериям.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
action destination-nat { off | pool <NAME> | netmap <ADDR/LEN> }
no action destination-nat
```

### Параметры

off – трансляция отключена. Трафик, попадающий под заданные критерии, не будет изменен;

pool <NAME> – имя пула, содержащего набор IP-адресов и/или TCP/UDP-портов. У трафика, попадающего под заданные критерии, будет изменен IP-адрес и TCP/UDP-порт получателя на значения, выбранные из пула;

`netmap <ADDR/LEN>` – IP-подсеть, используемая при трансляции. У трафика, попадающего под заданные критерии, будет изменен IP-адрес получателя на IP-адрес из указанной подсети. Параметр задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-DNAT-RULE

### Пример

```
esr(config-dnat-rule)# action destination-nat netmap 10.10.10.0/24
```

### action source-nat

Данной командой назначается тип действия «трансляция адреса и порта отправителя» и параметры трансляции для трафика, удовлетворяющего критериям, заданным командами «match».

Использование отрицательной формы команды (`no`) устанавливает значение по умолчанию.

### Синтаксис

```
action source-nat { off | pool <NAME> | netmap <ADDR/LEN> [static] | interface  
[FIRST_PORT – LAST_PORT] }
```

```
no action source-nat
```

### Параметры

`off` – трансляция отключена. Трафик, попадающий под заданные критерии, не будет изменен;

`pool <NAME>` – задаёт пул IP-адресов и/или TCP/UDP-портов. У трафика, попадающего под заданные критерии, будет изменен IP-адрес и/или TCP/UDP-порт отправителя на значения, выбранные из пула;

`netmap <ADDR/LEN>` – задаёт IP-подсеть для трансляции. У трафика, попадающего под заданные критерии, будет изменен IP-адрес отправителя на IP-адрес из указанной подсети. Параметр задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32];

`static` – включение статического NAT, доступно при использовании `netmap`;

`interface [FIRST_PORT – LAST_PORT]` – задаёт трансляцию в IP-адрес интерфейса. У трафика, попадающего под заданные критерии, будет изменён IP-адрес отправителя на IP-адрес интерфейса, в который трафик будет отправлен. Если дополнительно задан диапазон TCP/UDP-портов, то трансляция будет происходить ещё и для TCP/UDP-портов отправителя, они будут заменены на указанный диапазон портов.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-SNAT-RULE

**Пример**

```
esr(config-snat-rule)# action source-nat netmap 10.10.10.0/24
```

**check output-interface**

Данной командой разрешается очистка SNAT-сессий, в случае если выходной интерфейс изменен. Использование отрицательной формы команды (no) удаляет назначенное действие.

**Синтаксис**

```
check output-interface  
no check output-interface
```

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-SNAT-RULE

**Пример**

```
esr(config-snat-rule)# check output-interface
```

**description**

Данной командой задаётся описание.

Использование отрицательной формы команды (no) удаляет описание.

**Синтаксис**

```
description <DESCRIPTION>  
no description
```



### Параметры

<DESCRIPTION> – описание, задаётся строкой до 255 символов.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-DNAT-RULESET

CONFIG-SNAT-RULESET

CONFIG-DNAT-RULE

CONFIG-SNAT-RULE

CONFIG-DNAT-POOL

CONFIG-SNAT-POOL

### Пример

```
esr(config-snat-ruleset)# description "test ruleset"
```

### enable

Данной командой активируется конфигурируемое правило.

Отрицательная форма команды (no) деактивирует использование правила.

### Синтаксис

[no] enable

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-DNAT-RULE

CONFIG-SNAT-RULE

## Пример

```
esr(config-snat-rule)# enable
```

## from

Данной командой ограничивается область применения группы правил. Правила будут применяться только для трафика, идущего из определенной зоны или интерфейса.

Использование отрицательной формы команды (no) удаляет ограничение области применения группы правил.

## Синтаксис

```
from { zone <NAME> | interface <IF> | tunnel <TUN> | default }
no from
```


## Параметры

<NAME> – имя зоны изоляции, задается строкой до 12 символов.

<IF> – имя интерфейса устройства, задается в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<TUN> – имя туннеля устройства, задается в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#);

default – обозначает группу правил для всего трафика, источник которого не попал под критерии других групп правил.

 Группа правил со значением «default» параметра «from» может быть только одна.

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-DNAT-RULESET

## Пример

```
esr(config-dnat-ruleset)# from zone untrusted
```

## ip address

Данной командой устанавливается внутренний IP-адрес, на который будет заменяться IP-адрес получателя.

Использование отрицательной формы команды (no) удаляет заданный IP-адрес.

### Синтаксис

```
ip address <ADDR>
```

```
no ip address
```

### Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-DNAT-POOL

### Пример

```
esr(config-dnat-pool)# ip address 10.10.10.10
```

### ip address-range

Данной командой задаётся диапазон внешних IP-адресов, на которые будет заменяться IP-адрес отправителя.

Использование отрицательной формы команды (no) удаляет заданный диапазон адресов.

### Синтаксис

```
ip address-range <IP>[-<ENDIP>]
```

```
no ip address-range
```

### Параметры

<IP> – IP-адрес начала диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<ENDIP> – IP-адрес конца диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Если не указывать IP-адрес конца диапазона, то в качестве IP-адреса для трансляции используется только IP-адрес начала диапазона.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-SNAT-POOL

## Пример

```
esr(config-snat-pool)# ip address-range 10.10.10.1-10.10.10.20
```

## ip nat proxy-arp

Данная команда позволяет маршрутизатору отвечать на ARP-запросы IP-адресов из указанного пула. Функция необходима для того, чтобы не назначать все IP-адреса из пула трансляции на интерфейс.

### Синтаксис

```
ip nat proxy-arp <OBJ-GROUP-NETWORK-NAME>  
no ip nat proxy-arp
```

### Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа.

### Значение по умолчанию

Функция NAT Proxy ARP отключена.

### Необходимый уровень привилегий

10

### Командный режим

```
CONFIG-IF-IG  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-CELLULAR-MODEM  
CONFIG-LT
```

## Пример

```
esr(config-if-gi)# ip nat proxy-arp nat-pool
```

## ip port

Данной командой устанавливается внутренний TCP/UDP-порт, на который будет заменяться TCP/UDP-порт получателя.

Использование отрицательной формы команды (no) удаляет заданный TCP/UDP-порт.

### Синтаксис

```
ip port <PORT>  
no ip port
```

### Параметры

<PORT> – TCP/UDP-порт, принимает значения [1..65535].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-DNAT-POOL

### Пример

```
esr(config-dnat-pool)# ip port 5000
```

## ip port-range

Данной командой задаётся диапазон внешних TCP/UDP-портов, на которые будет заменяться TCP/UDP-порт отправителя.

Использование отрицательной формы команды (no) удаляет заданный диапазон портов.

### Синтаксис

```
ip port-range <PORT>[-<ENDPORT>]  
no ip port-range
```

### Параметры

<PORT> – TCP/UDP-порт начала диапазона, принимает значения [1..65535];

<ENDPORT> – TCP/UDP-порт конца диапазона, принимает значения [1..65535]. Если не указывать TCP/UDP-порт конца диапазона, то в качестве TCP/UDP-порта для трансляции используется только TCP/UDP-порт начала диапазона.

### Необходимый уровень привилегий

10

## Командный режим

### CONFIG-SNAT-POOL

#### Пример

```
esr(config-snat-pool)# ip port-range 20-100
```

#### match destination-address

Данной командой устанавливается профиль IP-адресов получателя, для которых должно срабатывать правило.

При использовании команды «not» правило будет срабатывать для IP-адресов получателя, которые не входят в указанный профиль. Использование отрицательной формы команды (no) отменяет установленное действие.

#### Синтаксис

```
match [not] destination-address { address-range { <ADDR>[-<ADDR>] | <IPv6-ADDR>[-<IPv6-ADDR>] } | prefix { <ADDR/LEN> | <IPv6-ADDR/LEN> } | object-group <OBJ-GROUP-NETWORK-NAME> | any }
```

```
no match destination-address
```

#### Параметры

address-range <ADDR>[-<ADDR>] – диапазон IP-адресов для правил firewall. Если не указывать IP-адрес конца диапазона, то в качестве IP-адреса для срабатывания правила используется только IP-адрес начала диапазона.

Параметр задаётся в виде A.B.C.D, где каждая часть принимает значения [0..255]; <IPv6-ADDR> – IPv6-адрес, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

prefix <ADDR/LEN> – IP-подсеть, используемая для срабатывания правила фильтрации firewall.

Параметр задаётся в виде A.B.C.D/E, где каждая часть A – D принимает значения [0..255] и E принимает значения [1..32]; <IPv6-ADDR/LEN> – IPv6-адрес, задаётся в виде X:X:X::X/E, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и E принимает значения [1..128];

object-group <OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа.

При указании значения «any» правило будет срабатывать для любого IP-адреса получателя.

#### Значение по умолчанию

any

#### Необходимый уровень привилегий

10

## Командный режим

### CONFIG-DNAT-RULE

### CONFIG-SNAT-RULE

## Пример

```
esr(config-snat-rule)# match destination-address object-group remote
```

### match destination-address-port

Данной командой устанавливается профиль связок IP-адресов и TCP/UDP-портов получателя, для которых должно срабатывать правило.

При использовании параметра «not» правило будет срабатывать для связок IP-адресов и TCP/UDP-портов получателя, которые не входят в указанный профиль.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
match [not] destination-address-port { address-port { <ADDR>:<PORT> | <IPV6-ADDR>:<PORT> } | object-group <OBJ-GROUP-ADDRESS-PORT-NAME> | any }
```

```
no match destination-address
```

### Параметры

address-port { <ADDR>:<PORT> | <IPV6-ADDR>:<PORT> } – связка IP-адресов и TCP/UDP-портов. IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255], номер порта, принимает значение [1..65535]. <IPV6-ADDR>:<PORT> – IPv6-адрес, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF], номер порта, принимает значение [1..65535].

object-group <OBJ-GROUP-ADDRESS-PORT-NAME> – имя профиля связок IP-адресов и TCP/UDP-портов, задаётся строкой до 31 символа.

При указании значения «any» правило не будет учитывать данный способ фильтрации.

### Значение по умолчанию

any

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-SNAT-RULE

## Пример

```
esr(config-snat-rule)# match destination-address object-group local
```

### match destination-port

Данной командой устанавливается профиль TCP/UDP-портов получателя, для которых должно срабатывать правило.

При использовании команды «not» правило будет срабатывать для TCP/UDP-портов получателя, которые не входят в указанный профиль.

Использование отрицательной формы команды (no) отменяет установленное действие.

### Синтаксис

```
match [not] destination-port { port-range <PORT>[-<PORT>] | object-group <PORT-SET-NAME>
| any }
no match destination-port
```

### Параметры

port-range <PORT>[-<PORT>] – address-port <PORT>[-<PORT>] – диапазон TCP/UDP-портов для правил firewall. Если не указывать TCP/UDP-порт конца диапазона, то в качестве TCP/UDP-порта для срабатывания правила используется только порт начала диапазона.

<PORT-SET-NAME> – имя профиля TCP/UDP-портов, задаётся строка до 31 символа.

При указании значения «any» правило будет срабатывать для любого TCP/UDP-порта получателя.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-DNAT-RULE  
CONFIG-SNAT-RULE

### Пример

```
esr(config-snat-rule)# match destination-port object-group ssh
```

### match icmp

Данная команда используется для настройки параметров протокола ICMP, если он выбран командой «match protocol». Командой устанавливается тип и код сообщений протокола ICMP, для которых должно срабатывать правило.

При использовании команды «not» правило будет срабатывать для всех типов и кодов сообщений протокола ICMP, кроме указанных.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
match [not] icmp { <ICMP_TYPE> <ICMP_CODE> | <OPTION> }
no match icmp
```

### Параметры

<ICMP\_TYPE> – тип сообщения протокола ICMP, принимает значения [0 ..255];

<ICMP\_CODE> – код сообщения протокола ICMP, принимает значения [0 ..255]. При указании значения «any» правило будет срабатывать для любого кода сообщения протокола ICMP;



<OPTION> – стандартные типы ICMP-сообщений, может принимать значения:

- administratively-prohibited;
- alternate-address;
- conversion-error;
- dod-host-prohibited;
- dod-network-prohibited;
- echo;
- echo-reply;
- host-isolated;
- host-precedence;
- host-redirect;
- host-tos-redirect;
- host-tos-unreachable;
- host-unknown;
- host-unreachable;
- information-reply;
- information-request;
- mask-reply;
- mask-request;
- network-redirect;
- network-tos-redirect;
- network-tos-unreachable;
- network-unknown;
- network-unreachable;
- option-missing;
- packet-too-big;
- parameter-problem;
- port-unreachable;
- precedence;
- protocol-unreachable;
- reassembly-timeout;
- router-advertisement;
- router-solicitation;
- source-quench;
- source-route-failed;
- time-exceeded;
- timestamp-reply;
- timestamp-request;
- traceroute.

### **Значение по умолчанию**

any any

### **Необходимый уровень привилегий**

10

### **Командный режим**

CONFIG-DNAT-RULE

CONFIG-SNAT-RULE

## Пример

```
esr(config-snat-rule)# match icmp 2 any
```

## match protocol

Данной командой устанавливается имя или номер IP-протокола, для которого должно срабатывать правило.

При использовании параметра «not» правило будет срабатывать для всех протоколов, кроме указанного. Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
match [not] protocol <TYPE>
```

```
no match protocol
```

```
match [not] protocol-id <ID>
```

```
no match protocol-id
```

## Параметры

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre.

При указании значения «any» правило будет срабатывать для любых протоколов;

<ID> – идентификационный номер IP-протокола, принимает значения [0x00-0xFF].

## Значение по умолчанию

any

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-DNAT-RULE

CONFIG-SNAT-RULE

## Пример

```
esr(config-snat-rule)# match protocol udp
```

## match source-address

Данной командой устанавливается профиль IP-адресов отправителя, для которых должно срабатывать правило.

При использовании команды «not» правило будет срабатывать для IP-адресов отправителя, которые не входят в указанный профиль.

Использование отрицательной формы команды (no) отменяет установленное действие.

## Синтаксис

```
match [not] source-address { address-range { <ADDR>[-<ADDR>] | <IPv6-ADDR>[-<IPv6-ADDR>] } | prefix { <ADDR/LEN> | <IPv6-ADDR/LEN> } | object-group <OBJ-GROUP-NETWORK-NAME> | any }
```

```
no match source-address
```

## Параметры

**address-range <ADDR>[-<ADDR>]** – диапазон IP-адресов для правил firewall. Если не указывать IP-адрес конца диапазона, то в качестве IP-адреса для срабатывания правила используется только IP-адрес начала диапазона.

Параметр задаётся в виде A.B.C.D, где каждая часть принимает значения [0..255]; <IPv6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

**prefix <ADDR/LEN>** – IP-подсеть, используемая для срабатывания правила фильтрации firewall.

Параметр задаётся в виде A.B.C.D/E, где каждая часть A – D принимает значения [0..255] и E принимает значения [1..32]; <IPv6-ADDR/LEN> – IPv6-адрес, задаётся в виде X:X:X:X::X/E, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и E принимает значения [1..128];

**object-group <OBJ-GROUP-NETWORK-NAME>** – имя профиля IP-адресов, задаётся строкой до 31 символа.

При указании значения «any» правило будет срабатывать для любого IP-адреса получателя.

## Значение по умолчанию

any

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-DNAT-RULE

CONFIG-SNAT-RULE

## Пример

```
esr(config-snat-rule)# match source-address object-group local
```

## match source-address-port

Данной командой устанавливается профиль связок IP-адресов и TCP/UDP-портов отправителя, для которых должно срабатывать правило.

При использовании параметра «not» (match not) правило будет срабатывать для связок IP-адресов и TCP/UDP-портов отправителя, которые не входят в указанный профиль.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
match [not] source-address-port { address-port { <ADDR>:<PORT> | <IPV6-ADDR>:<PORT> } |
object-group <OBJ-GROUP-ADDRESS-PORT-NAME> | any }
no match source-address-port <OBJ-GROUP-ADDRESS-PORT-NAME>
```

## Параметры

address-port { <ADDR>:<PORT> | <IPV6-ADDR>:<PORT> } – связка IP-адресов и TCP/UDP-портов. IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255], номер порта, принимает значение [1..65535]. <IPV6-ADDR>:<PORT> – IPv6-адрес, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF], номер порта, принимает значение [1..65535].

object-group <OBJ-GROUP-ADDRESS-PORT-NAME> – имя профиля связок IP-адресов и TCP/UDP-портов, задаётся строкой до 31 символа.

При указании значения «any» правило не будет учитывать данный способ фильтрации.

## Значение по умолчанию

any

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-SNAT-RULE

## Пример

```
esr(config-snat-rule)# match source-address-port object-group admin
```

## match source-port

Данной командой устанавливается профиль TCP/UDP-портов отправителя, для которых должно срабатывать правило.

При использовании команды «not» правило будет срабатывать для TCP/UDP-портов отправителя, которые не входят в указанный профиль. Использование отрицательной формы команды (no) отменяет установленное действие.

## Синтаксис

```
match [not] source-port { port-range <PORT>[-<PORT>] | object-group <PORT-SET-NAME> |
any }
no match source-port
```

## Параметры

port-range <PORT>[-<PORT>] – address-port <PORT>[-<PORT>] – диапазон TCP/UDP-портов для правил firewall. Если не указывать TCP/UDP-порт конца диапазона, то в качестве TCP/UDP-порта для срабатывания правила используется только порт начала диапазона.

<PORT-SET-NAME> – имя профиля TCP/UDP-портов, задаётся строка до 31 символа.

При указании значения «any» правило будет срабатывать для любого TCP/UDP-порта получателя.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-DNAT-RULE

CONFIG-SNAT-RULE

### Пример

```
esr(config-snat-rule)# match source-port object-group telnet
```

## nat alg

Данная команда включает функцию трансляции IP-адресов в заголовках уровня приложений.

Использование отрицательной формы команды (no) отключает функцию трансляции IP-адресов в заголовках уровня приложений.

### Синтаксис

```
[no] nat alg { <PROTOCOL> }
```

### Параметры

<PROTOCOL> – протокол уровня приложений, в заголовках которого должна работать трансляция адресов, принимает значения [ftp, h323, pptp, netbios-ns, gre, sip, tftp].

Вместо имени отдельного протокола можно использовать ключ "all", который включает трансляцию IP-адресов в заголовках всех доступных протоколов.

### Значение по умолчанию

Функция трансляции IP-адресов в заголовках уровня приложений отключена.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# nat alg ftp
```

## nat destination

Данная команда позволяет войти в режим настройки сервиса трансляции адресов получателя (DNAT, Destination NAT).

Использование отрицательной формы команды (no) удаляет настройки сервиса трансляции адресов получателя (DNAT, Destination NAT).

### Синтаксис

```
[no] nat destination
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# nat destination  
esr(config-dnat)#
```

## nat source

Данная команда позволяет войти в режим настройки сервиса трансляции адресов отправителя (SNAT, Source NAT).

Использование отрицательной формы команды (no) удаляет настройки сервиса трансляции адресов отправителя (SNAT, Source NAT).

### Синтаксис

```
[no] nat source
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

## Пример

```
esr(config)# nat source
esr(config-snat)#
```

## persistent

Командой выполняется включение функции NAT persistent.

NAT persistent позволяет приложениям использовать STUN (session traversal utilities for NAT – утилиты проброса сессий для NAT) для установления соединения с устройствами, находящимися за шлюзом NAT. При этом гарантируется, что запросы от одного и того же внутреннего адреса транслируются в один и тот же внешний адрес.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
[no] persistent
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Функция NAT persistent отключена.

## Необходимый уровень привилегий

10

## Командный режим


CONFIG-SNAT-POOL

## Пример

```
esr(config-snat-pool)# persistent
```

## pool

Команда создаёт и назначает пул IP-адресов и TCP/UDP-портов с определённым именем для сервиса NAT и меняет командный режим на SNAT POOL или DNAT POOL.

 Если пул используется в какой-либо группе правил, то его удалять нельзя.

Использование отрицательной формы команды (no) удаляет заданный пул NAT-адресов.

## Синтаксис

```
[no] pool <NAME>
```

## Параметры

<NAME> – имя пула NAT-адресов, задаётся строкой до 31 символа. Если использовать команду для удаления, то при указании значения «all» будут удалены все пулы IP-адресов и TCP/UDP-портов.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-DNAT

CONFIG-SNAT

## Пример

```
esr(config-snat)# pool nat
esr(config-snat-pool)#
```

## rearrange

Данная команда меняет шаг между созданными правилами.

## Синтаксис

```
rearrange <VALUE>
```

## Параметры

<VALUE> – шаг между правилами, принимает значения [1..50].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-DNAT-RULESET

CONFIG-SNAT-RULESET

## Пример

```
esr(config-dnat-ruleset)# rearrange 10
```

## renumber rule

Данная команда меняет номер правила.



**Синтаксис**

```
renumber rule <CUR_ORDER> <NEW_ORDER>
```

**Параметры**

<CUR\_ORDER> – текущий номер правила, принимает значения [1..10000];

<NEW\_ORDER> – новый номер правила, принимает значения [1..10000].

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-DNAT-RULESET

CONFIG-SNAT-RULESET

**Пример**

```
esr(config-dnat-ruleset)# renumber rule 13 100
```

**rule**

Данной командой создается правило с определённым номером и устанавливается режим командного интерфейса SNAT RULE или DNAT RULE. Правила обрабатываются устройством в порядке возрастания номеров правил.

Использование отрицательной формы команды (no) удаляет правило по номеру либо все правила.

**Синтаксис**

```
[no] rule <ORDER>
```

**Параметры**

<ORDER> – номер правила, принимает значения [1..10000]. Если использовать команду для удаления, то при указании значения «all» будут удалены все правила.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-DNAT-RULESET

CONFIG-SNAT-RULESET

**Пример**

```
esr(config-snat-ruleset)# rule 10
```

```
esr(config-snat-rule)#
```

## ruleset

Данная команда используется для создания группы правил с определённым именем и перехода в командный режим SNAT RULESET или DNAT RULESET.

Использование отрицательной формы команды (no) удаляет заданную группу правил.

### Синтаксис

```
[no] ruleset <NAME>
```

### Параметры

<NAME> – имя группы правил, задаётся строкой до 31 символа. Если использовать команду для удаления, то при указании значения «all» будут удалены все группы правил.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-DNAT

CONFIG-SNAT

### Пример

```
esr(config-snat)# ruleset wan  
esr(config-snat-ruleset)#
```

## show ip nat alg

Данная команда используется для просмотра информации о функционале трансляции IP-адресов в заголовках уровня приложений.

### Синтаксис

```
show ip nat alg
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

## Пример

```

esr# show ip nat alg
ALG Status:
  FTP:    Enabled
  H.323:  Disabled
  GRE:    Disabled
  PPTP:   Disabled
  SIP:    Disabled
  SNMP:   Disabled
  TFTP:   Disabled

```

## show ip nat pool

Данная команда используется для просмотра пулов внутренних и внешних IP-адресов и TCP/UDP-портов.

### Синтаксис

```
show ip nat <TYPE> pools
```

### Параметры

<TYPE> – тип пулов, для просмотра:

- source – внешние IP-адреса и TCP/UDP-порты;
- destination – внутренние IP-адреса и TCP/UDP-порты.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

## Пример

```

esr# show nat source pools
  Pools
  ~~~~~
ID      Name                Ip address          Port range          Description          Persi
-----  -----
0       outside                25.56.48.11        2000 - 3000        outside-pool       false

```

## show ip nat ruleset

Данной командой выполняется просмотр всех или выбранных групп правил, используемых функцией NAT.

## Синтаксис

```
show ip nat <TYPE> ruleset [<NAME>]
```

## Параметры

<TYPE> – тип группы правил:

- source – группа правил для трансляции IP-адреса и TCP/UDP-порта отправителя;
- destination – группа правил для трансляции IP-адреса и TCP/UDP-порта получателя.

[NAME] – имя группы правил, опциональный параметр. Если имя не задано – будет выведен список всех групп правил.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```

esr# show ip nat source rulesets
Rulesets
~~~~~
ID      Name                               To                               Description
-----
0       factory                             zone 'untrusted'
1       test                                gigabitethernet
                                           1/0/1
esr# show ip nat source rulesets factory
Ruleset:          factory
Description:
To:              none
Rules:
-----
Order:           10
Description:     replace 'source ip' by outgoing interface ip address
Matching pattern:
  Protocol:      any(0)
  Src-addr:      any
  Dest-addr:     any
Action:          interface port any
Status:          Enabled
-----

```

## show ip nat translations

Данная команда используется для просмотра сессий трансляции. Для просмотра информации о статистике необходимо включить счетчики (раздел [ip firewall mode](#)).

## Синтаксис

```
show ip nat translations [ vrf <VRF> ] [ protocol <TYPE> ] [ inside-source-address <ADDR> ] [ outside-source-address <ADDR> ] [ inside-destination-address <ADDR> ]
```

```
[ outside-destination-address <ADDR> ] [ inside-source-port <PORT> ] [ outside-source-
port <PORT> ] [ inside-destination-port <PORT> ] [ outside-destination-port <PORT> ]
[ summary ]
```

## Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут отображены сессии трансляций в указанном VRF;

summary – выводит суммарную статистику по сессиям трансляции;

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<PORT> – TCP/UDP-порт, принимает значения [1..65535].

Для Source NAT:

- inside-source-address – ключ для указания IP-адреса источника до трансляции;
- inside-destination-address – ключ для указания IP-адреса назначения на входе в маршрутизатор;
- outside-source-address – ключ для указания IP-адреса источника после трансляции;
- outside-destination-address – ключ для указания IP-адреса назначения на выходе из маршрутизатора.
- inside-source-port – ключ для указания TCP/UDP-порта отправителя до трансляции;
- outside-source-port – ключ для указания TCP/UDP-порта отправителя после трансляции;
- inside-destination-port – ключ для указания TCP/UDP-порта назначения до трансляции;
- outside-destination-port – ключ для указания TCP/UDP-порта назначения после трансляции.

Для Destination NAT:

- inside-source-address – ключ для указания IP-адреса источника на выходе из маршрутизатора;
- inside-destination-address – ключ для указания IP-адреса назначения после трансляции;
- outside-source-address – ключ для указания IP-адреса источника на входе в маршрутизатор;
- outside-destination-address – ключ для указания IP-адреса назначения до трансляции;
- inside-source-port – ключ для указания TCP/UDP-порта отправителя до трансляции;
- outside-source-port – ключ для указания TCP/UDP-порта отправителя после трансляции;
- inside-destination-port – ключ для указания TCP/UDP-порта назначения до трансляции;
- outside-destination-port – ключ для указания TCP/UDP-порта назначения после трансляции.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример 1

Source NAT

```
esr# show ip nat translations
Prot  Inside source  Inside destination  Outside source  Outside destination  Pkts  Bytes
----  -
icmp  115.0.0.10     1.1.0.2             1.1.0.24       1.1.0.2             3     252
```

## Пример 2

### Destination NAT

```

esr# show ip nat translations
Prot  Inside source  Inside destination  Outside source  Outside destination  Pkts  Bytes
----  -
icmp  1.1.0.2        115.0.0.10         1.1.0.2        1.1.0.16           --    --

```

### show ip nat proxy-arp

Данная команда используется для просмотра настроек NAT Proxy ARP.

### Синтаксис

```
show ip nat proxy-arp
```

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```

esr# show nat proxy-arp
Interface      IP address range
-----
gi1/0/15      115.0.0.15-115.0.0.100

```

### to

Данной командой ограничивается область применения группы правил. Правила будут применяться только для трафика, идущего в определенную зону или интерфейс.

Использование отрицательной формы команды (no) удаляет ограничение области применения группы правил.

### Синтаксис

```

to { zone <NAME> | interface <IF> | tunnel <TUN> | default }
no to

```


### Параметры

<NAME> – имя зоны изоляции;

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#);

default – обозначает группу правил для всего трафика, место назначение которого не попало под критерии других групп правил.

 Группа правил со значением «default» параметра «to» может быть только одна.

### Значение по умолчанию

None

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-SNAT-RULESET

### Пример

```
esr(config-snat)# ruleset test
esr(config-snat-ruleset)# to interface gigabitethernet 1/0/1
```

## 22 Управление тунелированием, VPN и удалённым доступом

- Генерация и просмотр ключей и сертификатов
- Конфигурирование и мониторинг туннелей
- Настройки IPsec VPN
- Управление VPN. Настройки удаленного доступа

### Генерация и просмотр ключей и сертификатов

- `crypto generate cert`
- `crypto generate csr`
- `crypto generate pfx`
- `crypto generate private-key`
- `crypto generate public-key`
- `show crypto certificates`
- `show crypto certificates cert`
- `show crypto certificates crl`
- `show crypto certificates dh`
- `show crypto certigicates pfx`
- `show crypto certificates private-key`
- `show crypto certificates public-key`
- `show crypto certificates ta`

#### **crypto generate cert**

Данной командой производится генерация сертификата x.509. Для генерации сертификата необходим файл запроса на подписание сертификата, сертификат удостоверяющего центра (CA), а также приватный ключ.

#### **Синтаксис**

```
crypto generate cert csr <CSR> ca <CA> private-key <PRIVATE-KEY> [[invalid-after <TIME>
<DAY> <MONTH> <YEAR>][valid-after <TIME> <DAY> <MONTH> <YEAR>]] filename <NAME>
```

#### **Параметры**

<CSR> – имя файла запроса на подписание сертификата;

<CA> – сертификат удостоверяющего центра (CA);

<PRIVATE-KEY> – приватный ключ;

invalid-after – параметр задаёт дату и время, до которых сертификат считается действительным;

valid-after – данный параметр задаёт дату и время, начиная с которых сертификат считается действительным;

<TIME> – время, указанное в формате ЧЧ:ММ:СС;

<DAY> – день месяца в диапазоне от 1 до 31. Если в месяце нет такого числа, то команда не отработает;

<YEAR> – год в диапазоне от 1970 до 2100 года;

<NAME> – имя сертификата x.509.

#### **Необходимый уровень привилегий**

15



## Командный режим

ROOT

### Пример

```
esr# crypto generate cert csr req.pem ca ca.cert private-key private.key filename cert.crt
```

### crypto generate csr

Данной командой производится генерация файла запроса на подписание сертификатов. Для генерации запроса необходимо наличие приватного ключа.

### Синтаксис

```
crypto generate csr private-key <PRIVATE-KEY> [[alternative-name <ALT-NAME>][common-name <COMMON-NAME>][country <COUNTRY>][email-address <E-MAIL>][locality <CITY>][organization <ORG-NAME>][organizational-unit <ORG-UNIT-NAME>][state <STATE>]] filename <NAME>
```

### Параметры

- <PRIVATE-KEY> – имя приватного ключа, задаётся строкой до 31 символа;
- <ALT-NAME> – альтернативное имя, задаётся строкой от 5 до 255 символов;
- <COMMON-NAME> – общее название, задаётся строкой до 61 символа;
- <COUNTRY> – название страны, задаётся строкой до 2 символов;
- <E-MAIL> – адрес электронной почты, задаётся строкой от 3 до 64 символов;
- <CITY> – название населенного пункта, задаётся строкой до 128 символов;
- <ORG-NAME> – название субъекта организации, задаётся строкой до 64 символов;
- <ORG-UNIT-NAME> – название организационной единицы, задаётся строкой до 64 символов;
- <STATE> – название области, провинции или штата, задаётся строкой до 128 символов;
- <NAME> – имя файла запроса на подписание сертификатов.

### Необходимый уровень привилегий

15

## Командный режим

ROOT

### Пример

```
esr# crypto generate csr private-key erver.key common-name ec-client.esr.eltex.loc country RU
email-address esr@eltex.loc locality Novosibirsk organization Eltex-Co-Ltd organizational-unit
ESR state NSK filename csr.pem
```

## crypto generate pfx

Данной командой производится генерация контейнера PKCS12. Для создания контейнера необходим сертификат удостоверяющего центра (CA), сертификат x.509, выпущенный этим удостоверяющим центром и приватный ключ этого сертификата.

### Синтаксис

```
crypto generate pfx private-key <PRIVATE-KEY> cert <CERT> ca <CA> [password ascii-text <PASSWORD>] filename <NAME>
```

### Параметры

<PRIVATE-KEY> – имя приватного ключа;

<CERT> – имя сертификата;

<CA> – имя сертификата удостоверяющего центра (CA);

<PASSWORD> – пароль для PKCS12-контейнера;

<NAME> – имя генерируемого PKCS12-контейнера, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

15

### Командный режим

ROOT

### Пример

```
esr# crypto generate pfx private-key server.key cert server.crt ca ca.crt password ascii-text password filename gen_esr.p12
```

## crypto generate private-key

Данной командой производится генерация приватного ключа.

### Синтаксис

```
crypto generate private-key <KEY-TYPE> [<KEY-SIZE>] filename <NAME>
```

### Параметры

<KEY-TYPE> – тип генерируемого ключа:

- rsa – ключ RSA;
- x25519 – ключ формата x25519, совместимый с Wireguard.

<KEY-SIZE> – размер ключа в битах. Значение может находиться в диапазоне от 1024 до 4096;

<NAME> – имя ключа, задаётся строкой до 31 символа. Возможна перезапись уже существующих файлов.

**Необходимый уровень привилегий**

15

**Командный режим**

ROOT

**Пример**

```
esr# crypto generate private-key rsa 1024 filename gen_private1.key
.....+-----+
+.-----+.-----+
++++++
```

**crypto generate public-key**

Данной командой производится генерация публичного ключа. Для генерации публичного ключа необходимо наличие приватного ключа.

**Синтаксис**

```
crypto generate public-key x25519 private-key <PRIVATE-KEY> filename <NAME>
```

**Параметры**

<PRIVATE-KEY> – имя приватного ключа;

<NAME> – имя публичного ключа, задаётся строкой до 31 символа.

**Необходимый уровень привилегий**

15

**Командный режим**

ROOT

**Пример**

```
esr# crypto generate public-key x25519 private-key server.key filename pubkey.pem
```

**show crypto certificates**

Данной командой выполняются просмотр количества сертификатов и ключей в энергонезависимой памяти устройства, необходимых для построения VPN (cert, crl, dh, private-key, public-key, pfx и ta).

**Синтаксис**

```
show crypto certificates
```

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

10

## Командный режим

ROOT

## Пример

```

esr# show crypto certificates
Type                Total
-----            -
cert                9
dh                  2
private-key         5
public-key          0
ta                  1
crl                  1
pfx                  0

```

## show crypto certificates cert

Данной командой выполняется просмотр содержимого X.509-сертификатов.

## Синтаксис

show crypto certificates cert [<NAME>]

## Параметры

<NAME> – имя сертификата или ключа выбранного типа, задается строкой до 31 символа.

При выполнении команды без параметра происходит отображение всех сертификатов типа cert.

## Необходимый уровень привилегий

10

## Командный режим

ROOT

## Пример

```

esr# show crypto certificates cert ca.crt
Version:                3
Serial:                 04:7B:45:DD:3F:0B:00:7F:8D:AB:57:D6:1B:C5:A6:56:C
0:E
8:

```

```

49:E0
Subject name:
  C(countryName):          RU
  ST(stateOrProvinceName): Siberia
  L(localityName):        Novosibirsk
  O(organizationName):    Eltex
  OU(organizationalUnitName): Eltex SC
  CN(commonName):         ca.test.loc
  emailAddress(emailAddress): ca@test.loc
Issuer name:
  C(countryName):          RU
  ST(stateOrProvinceName): Siberia
  L(localityName):        Novosibirsk
  O(organizationName):    Eltex
  OU(organizationalUnitName): Eltex SC
  CN(commonName):         ca.test.loc
  emailAddress(emailAddress): ca@test.loc
Validity period:
  Valid after:             2024-06-21 07:39:55
  Invalid after:          2034-06-19 07:39:55
Signature:
  Algorithm:               sha256WithRSAEncryption
  Value:                   92:37:D5:4F:3E:A3:93:B3:AF:B1:7A:40:08:9F:7C:80:25:0
5:
89:2D:F7:E5:8F:63:B6:ED:4E:A2:F3:26:E6:57:B6:F2:12:0
5:
1C:E5:42:1F:BC:69:99:F4:88:D4:2A:12:26:21:8E:7B:1D:E
6:
1F:D3:95:FF:63:7C:63:7E:D9:2A:98:BA:C2:ED:F9:01:4F:2
A:
A8:4F:7B:59:60:65:79:1A:37:99:72:F1:C4:65:85:DE:1B:B
B:
E8:0E:C5:F1:C8:95:AE:E6:7B:6C:5F:2C:93:50:51:CC:1B:C
0:
37:50:76:77:60:8B:29:CE:A2:FD:ED:1F:6B:51:CF:A4:67:D
C:
6C:07:0B:EF:80:58:00:C9:C5:72:1E:7D:E2:92:9B:27:2B:5
8:
FE:4B:41:2B:0C:1C:86:86:36:77:D7:99:B8:74:7C:F3:9A:E
2:
0F:83:9B:3C:CE:71:68:B9:07:08:CF:36:9B:5C:8E:18:F9:A
C:
AF:CB:DF:90:2A:2F:40:63:AB:DE:37:40:8F:B8:10:1F:86:1
6:
DC:80:73:42:8D:91:BC:C7:96:6B:42:86:20:AB:EC:43:B7:6
4:
2A:BE:57:9E:CE:E4:1F:A4:61:E1:04:0A:F8:65:AC:C3:2D:3
3:
F6:6F:97:D5:C4:11:6C:40:57:AD:92:4A:C2:01:31:3F:9C:9
5:
5E:F0:2A:65:A3:4A:67:EE:69:F5:A5:B7:6D:AA:99:4B:BE:9
1:
68:8A:CA:BA:BC:9A:10:9B:B6:2D:F7:AE:90:C8:88:A7:84:C
4:
B9:60:20:1B:9B:C3:BF:6C:86:8B:58:DF:62:01:15:F6:A9:7
9:
EE:A6:94:CD:8B:49:06:B5:31:C0:AF:75:BB:2B:45:11:20:3
0:
3E:98:DF:71:32:BB:50:16:13:46:E8:0D:18:4A:BB:76:E1:4
6:
61:9A:07:5E:5D:21:56:8D:81:A6:FA:7C:C3:EA:A2:28:E4:4
0:

```

```

A: 75:6C:2D:2A:42:57:66:5C:FE:3C:E3:54:88:B7:26:BC:15:8
6: CC:F0:1D:FF:22:A5:98:AC:2D:3E:6C:99:CF:98:5F:10:C4:0
6: D4:13:81:FF:79:2B:AE:24:81:09:CB:9C:FC:93:C3:46:96:B
E: BA:29:3D:7C:70:32:19:68:63:50:AB:E2:6C:CE:2E:69:65:D
A: 1B:B5:3A:B4:5C:9E:32:74:3D:96:5C:87:F1:3F:6C:55:73:F
6: 65:30:91:74:E7:5E:B6:11:61:89:A7:60:5F:51:E3:8E:11:4
2: D4:43:34:58:09:99:A3:55:A5:6E:44:56:DB:3C:82:B9:20:E
C: 1A:B2:E9:02:51:DB:39:30:54:28:1F:AE:73:C5:AE:7A:55:4
C4:BB:05:F1:0B:CA:60:8B
Public key info:
  Algorithm:          RSA
  Key size:           4096
  Exponent:           65537
  Modulus:            00:A2:CF:F0:E7:7A:19:0F:05:F5:6E:82:72:F0:AA:C6:2C:8
9: 01:F6:48:9E:79:13:ED:7D:4B:86:3D:22:FF:0E:9A:A1:7D:0
D: C6:69:85:C9:0E:6D:34:40:28:9F:E0:F0:9B:C5:C4:21:CA:A
7: 43:B5:1B:99:37:B1:F4:61:2F:1B:50:6E:8A:00:88:3D:4B:9
D: F2:D0:C0:C9:CB:E3:01:F2:D5:EB:64:C5:13:CB:F9:1F:47:1
2: 67:11:F9:F8:85:FB:B4:37:37:DA:80:B6:88:72:FE:9D:55:1
5: 31:DE:F5:C7:83:95:37:9E:0A:AD:B3:53:36:B3:1C:73:CE:5
A: 9B:BB:C5:63:11:13:8C:55:B9:54:43:FF:7F:57:60:83:7F:6
2: A0:76:CE:86:56:2C:0F:DD:8C:35:6C:BE:24:78:01:D5:77:F
4: 79:97:50:C5:D9:F1:BB:82:73:48:79:A8:08:DD:00:80:66:5
D: F5:DD:E3:FF:E1:FE:30:16:1A:59:74:56:12:74:92:B5:9C:2
8: FF:0B:4A:EE:E4:93:A8:B2:BF:5C:45:70:C5:59:61:9A:D0:7
6: F1:F3:83:31:67:77:48:A0:0D:58:3C:85:EA:55:AD:5A:7D:C
4: 7F:62:7D:70:BD:E3:8C:17:0D:AE:55:6F:28:FC:C9:6B:93:F
4: 92:E3:C2:FA:9F:99:32:DC:B7:E3:1E:F0:DF:6A:57:0F:AF:2
B: 11:4C:8F:17:C2:F0:CC:23:AD:A3:27:C6:E9:CC:90:50:21:7
9: A8:94:A3:2F:4C:5A:2F:AD:71:D3:1B:3C:A1:52:5F:76:E6:4
F: 63:2F:43:98:5E:36:CC:8C:FE:87:47:E4:9D:93:0B:71:99:A
8: DD:95:C7:61:F2:CC:3E:C9:46:99:67:2C:AD:CB:30:62:0F:3
9: 45:93:98:10:4B:83:FB:50:F0:1E:38:82:11:6E:9E:93:AC:E
3: 84:56:CA:61:79:6B:F3:60:84:53:BE:D1:27:7A:33:47:29:6

```

```

2:
2D:A8:1D:3E:0F:CF:EF:69:5C:FB:13:7B:99:C2:88:E8:0D:0
7:
DD:7F:DC:8F:3C:C9:DC:9F:A8:7F:B2:8B:29:50:E7:6F:17:A
0:
E4:73:02:4C:6E:6F:13:A9:C9:79:3B:E8:38:3A:79:44:0D:C
8:
9C:10:85:CE:52:E1:40:4B:5D:CE:5F:47:D0:D7:1B:AD:71:0
9:
A8:94:37:65:D2:A0:4A:69:3D:ED:92:81:1E:10:AA:8F:DC:6
B:
79:6F:1F:3A:D1:0C:EA:A1:22:00:05:A1:35:58:0E:6A:2E:C
7:
FB:4E:B7:A8:23:97:9E:5A:1B:15:CA:15:3A:1D:01:58:83:8
2:
6D:7D:AD:8F:94:BC:33:C0:17
X509v3 Basic Constraints:
  CA: Yes
  Critical: Yes
X509v3 Subject key identifier:
  ID: 1D:60:BA:51:B1:16:35:83:53:E1:D4:33:46:40:93:BB:05:0
8:
  6F:FB
  Critical: No
X509v3 Authority key identifier:
  ID: 1D:60:BA:51:B1:16:35:83:53:E1:D4:33:46:40:93:BB:05:0
8:
  6F:FB
  Critical: No
X509v3 Key Usage:
  Usage: Digital Signature
  Certificate Sign
  CRL Sign
  Critical: Yes

```

## show crypto certificates crl

Данной командой выполнятся просмотр содержимого списка отозванных сертификатов.

### Синтаксис

```
show crypto certificates crl [<NAME>]
```

### Параметры

<NAME> – имя сертификата или ключа выбранного типа, задается строкой до 31 символа.

При выполнении команды без параметра происходит отображение всех сертификатов типа crl.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

## Пример

```

esr# show crypto certificates crl ca_crl.pem
Version: 2
Issuer name:
  C(countryName): RU
  ST(stateOrProvinceName): Siberia
  L(localityName): Novosibirsk
  O(organizationName): Eltex
  OU(organizationalUnitName): Eltex SC
  CN(commonName): ca.test.loc
  emailAddress(emailAddress): ca@test.loc
Update info:
  Last update: 2024-06-24 08:32:32
  Next update: 2024-07-24 08:32:32
Signature:
  Algorithm: sha256WithRSAEncryption
  Value: 9A:93:8A:CE:59:91:7B:1B:19:59:72:0F:3B:A5:45:F3:D
8:17: A5:8B:5C:AF:15:C9:2A:BE:25:1E:A2:67:E1:22:52:24:4
5:C2: FD:EC:9A:ED:6E:C2:1B:00:D7:2B:25:FB:50:EE:7F:F9:9
F:23: FC:FA:57:3D:D3:1F:50:B7:04:40:22:9D:13:52:7B:B3:4
2:5D: 6E:13:AA:6A:55:E1:5A:43:56:CF:27:B1:BE:23:A6:F3:1
5:1A: DA:81:19:02:56:68:F7:31:6C:22:81:83:FC:0A:A0:7A:0
6:01: 57:2B:AB:93:73:01:55:B4:7C:DB:CF:45:D1:52:0A:E1:7
3:14: 56:F6:55:DB:76:FA:66:2A:B9:9C:04:BD:4C:66:CC:21:9
7:14: AB:FD:D7:A6:64:AB:D4:E1:4F:12:B5:32:83:C5:62:60:5
B:85: A8:09:5B:B8:A0:00:CA:94:04:70:13:CE:CA:BC:F2:68:0
1:25: 6A:63:5C:44:B4:33:92:B5:4E:26:CA:28:1D:BB:A8:54:3
4:DC: 23:80:A2:49:AE:2B:CF:97:D4:84:0F:0F:8D:21:04:65:9
1:DC: 83:19:F1:FB:BD:AF:01:6F:FA:09:CD:67:95:26:EE:68:6
9:72: 6D:40:BA:D7:2B:D0:D5:82:E7:D9:B4:76:30:DD:7B:F2:A
3:C0: D5:44:07:DD:8C:86:46:27:6A:07:CB:10:CB:31:F8:8A:9
9:9D: 82:EA:B5:A5:2E:FE:8F:0E:77:75:62:1B:FE:E8:B5:98:8
7:F9: 09:02:2B:3D:58:91:56:83:D2:E7:41:B6:1E:1C:0E:05:F
0:6E: DE:F3:6D:AD:1E:FB:10:81:E3:F0:28:5F:A6:42:C9:03:6
5:B6: DC:85:DD:F7:97:9B:83:5D:C5:E1:FA:F0:90:9B:67:A3:0
D:0F: CE:0C:8F:40:D8:09:F7:12:F1:A3:D4:6B:AF:5E:D0:4E:5
6:3C: 41:5C:38:5E:8B:32:CA:B9:61:DD:26:87:BB:0C:D2:B2:2
C:16:

```



```

9:DC: F7:80:A3:E5:CA:92:E7:4B:30:DF:1F:21:58:80:72:23:0
3:5A: AC:A9:16:0B:FF:51:C4:5F:FB:E6:ED:36:30:8C:59:80:8
D:0E: 3A:68:07:3D:B0:63:57:A2:EC:04:F1:B0:13:C1:E0:9C:8
F:B9: FC:50:D8:2E:6A:A2:BD:47:26:1D:BE:CB:34:6C:D4:85:C
A:E5: 47:11:BB:E4:B3:0C:C8:2C:25:8E:1D:25:B6:6C:51:4B:3
2:29: D2:50:5C:89:03:55:68:0F:7A:C6:F3:08:01:9C:47:9C:C
5:9E: BF:22:20:6B:44:0C:DE:A1:26:C6:E8:FD:0F:19:E8:DC:4
X509v3 Authority key identifier: 6D:07:E1:D5:E9:FC:FC:AA
  ID: 1D:60:BA:51:B1:16:35:83:53:E1:D4:33:46:40:93:BB:0
5:08: 6F:FB
      Critical: No
X509v3 CRL Number:
  Number: 4097
  Critical: No
Revoked certificates:
  Serial: 4097
  Revocation date: 2024-06-24 08:32:21

```

## show crypto certificates dh

Данной командой выполняется просмотр имеющихся ключей Диффи-Хеллмана.

### Синтаксис

```
show crypto certificates dh
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример

```

esr# show crypto certificates dh
File name
-----
default_dh.pem
dh.pem

```

## show crypto certificates pfx

Командой выполняются просмотр содержимого контейнера PKCS12.

### Синтаксис

```
show crypto certificates pfx [<NAME>] [password ascii-text <PASSWORD>]
```

### Параметры

<NAME> – имя сертификата или ключа выбранного типа, задается строкой до 31 символа.

<PASSWORD> – пароль, назначенный при генерации контейнера PKCS12. Максимальная длина строки до 64 символов.

При выполнении команды без параметра <NAME> происходит отображение всех сертификатов типа pfx.

В случае, если пароль не задан, то нужно определить только параметр <NAME>.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример

```

esr# show crypto certificates pfx show_pkcs_no_pass.p12
MAC:
  Algorithm:                               sha1
  Iteration:                               2048
  Length:                                   20
  Salt length:                             8
PKCS7 Encrypted data:
  Algorithm:
  pbewithSHA1And40BitRC2-CBC
  Iteration:                               2048

Certificate bag:

  X509 attributes:

    localKeyID                             3A:56:1C:D
F:6D:67:39:1B:C6:12:33:D2:29:B3:6F:B7:6A:B
1:
                                           62:25

  Version:                                 1

  Serial:                                  24:60:8C:4
A:B7:7E:D7:2B:11:AB:34:98:48:6A:45:66:AB:0
F:
                                           7C:80

```

```

Subject name:
    C(countryName):                RU
    ST(stateOrProvinceName):      Russia
    L(localityName):
Novosibirsk
    O(organizationName):          Eltex
Enterprise Ltd
    CN(commonName):                esr-20

Issuer name:
    C(countryName):                RU
    ST(stateOrProvinceName):      Russia
    L(localityName):
Novosibirsk
    O(organizationName):          Eltex
Enterprise Ltd
    CN(commonName):                Eltex
default certificate authority
    Validity period:
        Valid after:                1970-01-01
00:03:26
        Invalid after:              2069-12-07
00:03:26
    Signature:
        Algorithm:
sha256WithRSAEncryption
        Value:
AA:B4:6C:4A:48:E6:64:62:D9:2A:05:4D:3D:7B:F7:70:0D:6
5:
BB:85:D0:B
B:08:A3:75:EB:B9:B1:D1:89:B7:CD:6F:4C:55:7
A:
81:F3:7C:F
2:E2:DC:68:DD:ED:4B:05:EC:9B:5A:E9:DA:DF:A
D:
AA:28:0B:B
B:49:3E:E6:73:29:09:20:35:E4:A2:A8:64:10:1
A:
C8:9B:BB:3A:60:8F:10:65:E7:EE:99:E7:C1:D9:6E:4F:61:7
8:
1D:35:A4:B
D:15:7C:78:4F:6C:66:B9:4E:7A:C1:08:8F:BD:8
4:
9:EB:3A:0E:8E:FD:97:E8:2F:0E:D6:2D:DD:9F:8
2:

```

2:49:E9:F8:D4:D8:0A:16:D8:28:29:4F:9A:5A:36:	76:DA:FD:0
F:3C:89:6C:EE:CD:67:45:AB:F0:D9:1F:B9:FE:8E:	26:2A:64:D
2:C6:42:55:14:B9:A8:81:CE:ED:E4:9B:4B:9B:49:	0A:C3:73:3
2:FE:36:0D:1D:86:D1:18:20:CD:66:C6:18:50:7B:	21:1F:7E:2
6:A4:CB:E5:2A:E2:10:F0:A1:12:81:9A:BD:B1:7D:	71:85:04:4
8:BD:06:77:29:30:7C:21:FF:0F:98:F1:37:3D:CE:	42:84:91:E
B:12:D0:CA:74:85:DB:00:18:BC:FD:F0:1B:11:9B:	5F:DC:41:3
B	C2:21:22:2
Public key info:	
Algorithm:	RSA
Key size:	2048
Exponent:	65537
Modulus:	00:96:26:1
7:9E:E9:03:B9:03:43:16:6A:56:43:81:F2:FB:D F:	
D9:F6:A1:55:9A:3A:6D:8D:84:43:31:89:CD:22:B7:C8:4E:6 1:	49:E4:3C:A
0:95:3B:D4:CE:2B:93:58:5E:EE:22:26:7A:4C:A 2:	11:EA:12:8
7:79:7E:E0:CF:F9:BA:B3:71:DE:C4:83:4A:84:D 0:	50:7E:B0:9
4:EF:D3:49:B0:9F:EA:37:F5:B3:E4:10:15:C9:7 F:	
2B:B6:6C:83:A5:8E:BE:0B:E3:C3:59:1A:A3:18:B2:95:E7:3 0:	

```

7:39:48:6D:6A:FD:97:0E:17:71:BB:88:12:FE:2
0:
15:DC:33:C

6:9B:5E:BC:0A:F6:73:BF:BE:45:17:F2:30:C4:E
3:
B8:EC:96:0

3:03:3E:3E:DC:3E:4C:EC:3D:9F:13:C7:57:56:A
2:
2E:F6:87:D

2:D3:A3:AD:FE:AC:17:87:5D:D2:0B:7C:88:3F:6
A:
BF:06:DD:1

9:BF:C1:07:9C:ED:8B:98:D7:43:C7:D7:9F:80:B
D:
39:24:E9:A

2:AC:0C:A7:AD:AA:B0:2D:C9:8B:38:3C:A9:70:E
D:
ED:9B:24:B

5F:1E:C0:14:24:B5:CB:EF:CF:3C:46:63:A2:D1:97:EB:48:0
9:

BD:3C:1E:A8:D2:A2:91:25:6A:14:6C:BF:57:A1:E7:CB:67:2
C:

7:C1
Certificate bag:

Version:
3

Serial:
44:55:FB:9
A:76:2A:2A:44:EA:90:1F:43:5F:B3:44:5C:44:9
C:

2A:02

Subject name:

C(countryName):
RU

ST(stateOrProvinceName):
Russia

L(localityName):
Novosibirsk

O(organizationName):
Eltex

Enterprise Ltd

CN(commonName):
Eltex
default certificate authority
Issuer name:

C(countryName):
RU

ST(stateOrProvinceName):
Russia
    
```

```

L(localityName):
Novosibirsk
O(organizationName):                               Eltex
Enterprise Ltd
CN(commonName):                                    Eltex
default certificate authority
Validity period:

Valid after:                                       1970-01-01
00:03:24
Invalid after:                                    2069-12-07
00:03:24
Signature:

Algorithm:
sha256WithRSAEncryption
Value:                                             95:42:AC:6
2:99:71:D8:86:53:13:E6:90:60:21:40:48:99:3
F:

8B:BC:F9:9D:A7:DC:50:1B:18:97:12:DB:75:88:5A:19:68:3
5:

6:40:04:18:D2:95:45:68:25:29:39:CD:FC:03:2
E:

B2:BE:DA:E7:A0:09:92:A2:11:E1:3E:63:E0:D5:16:8D:4C:A
7:

E:77:11:AD:5C:14:01:A2:99:9B:24:F2:BF:34:B
0:

C:A5:D1:B8:32:77:83:0F:4B:EF:AD:E3:8F:97:5
0:

C:EB:98:E9:CC:4A:36:AC:0A:3E:32:6B:13:4B:8
8:

8:65:84:62:86:FD:6F:E0:71:8D:18:A3:5F:E8:C
3:

C:2E:1D:A4:B8:5C:35:A9:BD:BA:76:A2:AA:58:8
F:

1:A7:A0:CE:49:67:DD:C8:4B:0F:A0:CF:19:F5:E
7:

BB:A5:D1:8F:68:44:35:0E:49:BC:D1:DE:C6:F7:1C:8B:43:7
C:

4:F7:FB:23:FC:C8:4E:32:B6:76:CA:C6:85:0D:F
20:12:04:1

```

```

6:
                                                                 C7:61:8F:3
2:AF:B0:D1:4C:2B:89:83:C4:2F:5D:85:C7:AC:B
B:
                                                                 25:4F:22:6
F:7B:28:25:A8:0E:2B:74:CC:E4:77:C4:F3:F9:9
D:
                                                                 E7:F9:68:D
B
  Public key info:
    Algorithm: RSA
    Key size: 2048
    Exponent: 65537
    Modulus: 00:AD:FA:0
4:70:35:30:4A:8A:67:32:72:09:DC:D6:6F:2B:6
3:
                                                                 A3:BB:35:D
C:7A:17:6D:3C:84:AA:24:A8:7C:DE:0F:21:B7:0
6:
                                                                 35:AC:C7:6
F:91:34:B4:04:EB:99:67:15:3B:A6:A0:E7:F6:0
3:
                                                                 EF:D5:5D:38:63:3A:52:4B:23:50:68:03:B7:16:A9:99:D4:2
D:
                                                                 2A:54:98:2
D:98:7C:9F:D9:54:C3:44:C1:6A:09:EF:2A:21:D
5:
                                                                 55:A4:0B:6
3:65:95:CD:4A:75:6A:AC:44:A4:C8:3F:D0:A5:B
8:
                                                                 AA:F6:03:0
1:6C:94:B7:26:FC:AC:CE:4E:3A:B1:0B:B6:E7:F
8:
                                                                 66:3B:FA:B
1:DA:07:61:77:D5:0C:38:80:CF:6C:38:B3:8E:6
A:
                                                                 1B:76:A7:1
0:AE:35:41:B7:C5:48:FA:21:27:68:8A:4D:BD:4
9:
                                                                 E4:61:9B:1
1:D9:4F:F2:A1:0B:AF:49:53:07:C8:49:21:E1:D
3:
                                                                 56:09:85:7
7:72:DC:A6:EB:4C:C5:C2:A6:2C:38:5B:79:34:F

```

```

5:
3:6C:16:7E:89:E5:FC:11:AB:D6:C8:03:B4:05:C
F:
9:62:61:58:07:67:52:72:57:4D:28:C5:0D:E4:D
B:
E:D3:FF:55:E2:21:53:FF:0F:1F:AB:D0:7A:23:6
A:
A:29
    X509v3 Subject key identifier:
        ID:
6:A2:F0:2F:11:E6:18:48:DE:FF:C9:A4:27:54:5
B:
        Critical:
        X509v3 Authority key identifier:
            ID:
6:A2:F0:2F:11:E6:18:48:DE:FF:C9:A4:27:54:5
B:
            Critical:
            X509v3 Basic Constraints:
                CA:
                Critical:
PKCS7 Data:
    Shrouded Keybag:
        X509 attributes:
            localKeyID
F:6D:67:39:1B:C6:12:33:D2:29:B3:6F:B7:6A:B
1:
        Algorithm:
pbewithSHA1And3-KeyTripleDES-CBC
        Iteration:
        Key algorithm:
        Key size:

```

03:3A:16:C

BF:15:A5:C

15:95:A4:0

21:8D:47:3

03:CC:9D:E

3F:D7

No

03:CC:9D:E

3F:D7

No

Yes

Yes

3A:56:1C:D

62:25

2048

RSA

2048



## show crypto certificates private-key

Данной командой выполняется просмотр содержимого приватного ключа.

### Синтаксис

```
show crypto certificates private-key [<NAME>]
```

### Параметры

<NAME> – имя сертификата или ключа выбранного типа, задается строкой до 31 символа.

При выполнении команды без параметра происходит отображение всех сертификатов типа private-key.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример

```
esr# show crypto certificates private-key server.key
Key info:
File name:                server.key
Issuer:                   Private key: (2048 bit)
```

## show crypto certificates public-key

Командой выполняется просмотр содержимого публичного ключа.

### Синтаксис

```
show crypto certificates public-key [<NAME>]
```

### Параметры

<NAME> – имя сертификата или ключа выбранного типа, задается строкой до 31 символа.

При выполнении команды без параметра происходит отображение всех сертификатов типа public-key.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

## Пример

```
esr# show crypto certificates public-key pubkey.pem
File name:          pubkey.pem
Algorithm:          -
Length:             256
Location:           -
Fingerprint:       -
```

## show crypto certificates ta

Командой выполняются просмотр имеющихся HMAC-ключей.

## Синтаксис

```
show crypto certificates ta
```

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

10

## Командный режим

ROOT

## Пример

```
esr# show crypto certificates ta
File name
-----
ta.key
```

## Конфигурирование и мониторинг туннелей

- `auth-nocache`
- `authentication algorithm`
- `authentication method`
- `authentication required disable`
- `crypto`
- `crypto pfx`
- `clear ip nhrp peers`
- `clear ip nhrp shortcut-routes`
- `clear tunnels counters`
- `clear tunnels softgre`
- `compression`
- `default-profile`
- `description`
- `dscp`
- `enable`
- `encryption algorithm`
- `history statistics`
- `ignore-default-route`
- `interface`
- `ip dont-fragment-bit ignore`
- `ip nhrp attribute group`
- `ip nhrp authentication`
- `ip nhrp enable`
- `ip nhrp holding-time`
- `ip nhrp ipsec`
- `ip nhrp map`
- `ip nhrp map group`
- `ip nhrp multicast`
- `ip nhrp nhs`
- `ip nhrp redirect`
- `ip nhrp shortcut`
- `ip path-mtu-discovery discovery disable`
- `ip tcp adjust-mss`
- `ipsec authentication method`
- `ipsec authentication pre-shared-key`
- `ipsec ike proposal`
- `ipsec proposal`
- `keepalive dhcp dependent-interface`
- `keepalive dhcp link-timeout`
- `keepalive dst-address`
- `keepalive enable`
- `keepalive retries`
- `keepalive timeout`
- `keepalive timeout ipsec`
- `key`
- `load-average`
- `local address`
- `local address xauth`
- `local checksum`
- `local cookie`
- `local interface`
- `local port`
- `local session-id`
- `min-frag-size`

- mode
- mode
- mtu
- multipoint
- password
- peer
- peer lt
- port
- port
- ppp failure-count
- ppp timeout keepalive
- protocol
- private-key
- protocol
- public-key
- remote address
- remote address
- remote address xauth
- remote checksum
- remote cookie
- remote port
- remote session-id
- route-metric
- route-nopull
- snmp init-trap
- show ip nhrp peers
- show ip nhrp shortcut-routes
- show tunnels configuration
- show tunnels counters
- show tunnels history
- show tunnels status
- show tunnels utilization
- subnet
- ttl
- tunnel
- tunnel
- tunnel-source
- username
- username

### **auth-nocache**

Данной командой отключается кэширование пароля пользователя OpenVPN-клиента.

Использование отрицательной формы команды (no) активирует кэширование пароля пользователя OpenVPN-клиента.

### **Синтаксис**

[no] auth-nocache

### **Параметры**

Отсутствуют.

**Значение по умолчанию**

Кэширование разрешено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-OPENVPN

**Пример**

```
esr(config)# auth-nocache
```

**authentication algorithm**

Данной командой устанавливается алгоритм аутентификации, который используется для аутентификации при подключении к OpenVPN-серверу.

Использование отрицательной формы команды (no) удаляет алгоритм аутентификации.

**Синтаксис**

```
authentication algorithm <ALGORITHM>
```

```
no authentication algorithm
```

**Параметры**

<ALGORITHM> – алгоритм аутентификации, принимает значения: md4, rsa-md4, md5, rsa-md5, mdc2, rsa-mdc2, sha, sha1, rsa-sha, rsa-sha1, rsa-sha1-2, dsa, dsa-sha, dsa-sha1, dsa-sha1-old, ripemd160, rsa-ripemd160, ecdsa-with-sha1, sha-224, rsa-sha-224, sha-256, rsa-sha-256, sha-384, rsa-sha-384, sha-512, rsa-sha-512, whirlpool.

**Значение по умолчанию**

Не задано.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-OPENVPN

**Пример**

```
esr(config-openvpn)# authentication algorithm md5
```

## authentication method

Данной командой определяется метод аутентификации, который будет использоваться при установлении удаленного подключения клиентами PPPoE, PPTP и L2TP.

Использование отрицательной формы команды (no) удаляет указанный метод аутентификации.

### Синтаксис

```
authentication method <METHOD>  
no authentication method <METHOD>
```

### Параметры

<METHOD> – метод аутентификации, возможные значения: chap, mschap, mschap-v2, eap, pap.

### Значение по умолчанию

chap

### Необходимый уровень привилегий

10

### Командный режим

```
CONFIG-PPPOE  
CONFIG-PPTP  
CONFIG-L2TP
```

### Пример

```
esr(config-pppoe)# authentication method mschap-v2
```

## authentication required disable

Данной командой отключается обязательная аутентификация PPP на сервере PPTP и L2TP.

Использование отрицательной формы команды (no) включает аутентификацию PPP на сервере PPTP и L2TP.

### Синтаксис

```
[no] authentication required disable
```

### Параметры

Отсутствуют.

### Значение по умолчанию

Не задан.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-PPPOE  
 CONFIG-PPTP  
 CONFIG-L2TP

**Пример**

```
esr(config-pppoe)# authentication required disable
```

**crypto**

Данной командой указываются необходимые сертификаты для подключения к OpenVPN-серверу. Использование отрицательной формы команды (no) удаляет название сертификата из конфигурации.

**Синтаксис**

```
crypto <CERTIFICATE-TYPE> <NAME>  

no crypto <CERTIFICATE-TYPE>
```

**Параметры**

<CERTIFICATE-TYPE> – тип сертификата или ключа, может принимать следующие значения:

- ca – сертификат центра сертификации;
- cert – сертификат клиента;
- private-key – клиентский ключ;
- crl – список отозванных сертификатов;
- dh – ключ Диффи-Хеллмана;
- ta – HMAC-ключ.

<NAME> – имя сертификата или ключа, задаётся строкой до 31 символа.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-OPENVPN

## Пример

```
esr(config-openvpn)# crypto ca KEY
```

## crypto pfx

Данной командой указывается контейнер PKCS12.

Контейнер в обязательном порядке должен включать в себя сертификат удостоверяющего центра, сертификат x.509, выпущенный этим удостоверяющим центром и приватный ключ этого сертификата. Контейнер может содержать цепочку из сертификатов промежуточных удостоверяющих центров.

Использование команд **crypto ca**, **crypto cert**, **crypto private-key** и **crypto pfx** является взаимоисключающим.

## Синтаксис

```
crypto pfx <NAME> [password ascii-text <PASSWORD>]
no crypto pfx <NAME>
```

## Параметры

<NAME> – имя PKCS12-контейнера, задаётся строкой до 31 символа.

<PASSWORD> – пароль от PKCS12-контейнера.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-OPENVPN-SERVER

## Пример

```
esr(config-openvpn-server)# crypto pfx pkcs_openvpn_server.p12 password ascii-text password
```

## clear ip nhrp peers

Данная команда используется для очистки записей о NHRP-соседях с маршрутизатора.

## Синтаксис

```
clear ip nhrp peers [ { connected | group | lower-up | nat | nhs | protected | qos |
unique | used } ] [ { nbma-address <ADDR> | tunnel gre <ID> | tunnel-address <ADDR> } ]
[ type { cached | dynamic | incomplete | negative | static } ] [ vrf <VRF> ]
```

## Параметры

connected – очистить записи, для которых разрешена отправка пользовательских пакетов через GRE-туннель;

group – очистить записи, в которых NHRP-сосед сообщил свою NHRP-группу;



lower-up – очистить записи, в которых оперативное состояние GRE-туннеля UP;

nat – очистить записи, в которых NHRP-сосед расположен за NAT;

nhs – очистить записи, в которых осуществляется регистрация на NHRP-сервере;

protected – очистить записи, в которых GRE-туннель до NHRP-соседа защищен шифрованием IPsec;

qos – очистить записи, в которых к GRE-туннелю до NHRP-соседа применена QoS-политика на основании NHRP-группы;

unique – очистить записи, для которых требуется уникальность соответствия туннельного и NBMA-адреса;

used – очистить записи, в которых через GRE-туннель проходит пользовательский трафик;

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<ID> – идентификатор GRE-туннеля;

cached – очистить записи, соответствующие GRE-туннелям между NHRP-клиентами;

dynamic – очистить записи, соответствующие GRE-туннелям от NHRP-клиента до NHRP-сервера;

incomplete – очистить незавершенные записи, в которых известен туннельный адрес, но не определен NBMA-адрес;

negative – очистить ошибочные записи, для которых ни один доступный NHRP-сервер не имеет записи о запрошенном туннельном адресе;

static – очистить записи, заданные через конфигурацию маршрутизатора;

<VRF> – имя экземпляра VRF, задается строкой от 1 до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример

```
esr# clear ip nhrp peers tunnel-address 172.16.0.11
```

### clear ip nhrp shortcut-routes

Данная команда служит для очистки созданных shortcut-route записей до подсетей, находящихся за NHRP-клиентами.

### Синтаксис

```
clear ip nhrp shortcut-routes [ { network <ADDR/LEN> | nexthop <ADDR> | tunnel gre
<ID> } ] [ vrf <VRF> ]
```

### Параметры

<ADDR/LEN> – IP-подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32];

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<ID> – идентификатор GRE-туннеля;

<VRF> – имя экземпляра VRF, задается строкой от 1 до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример

```
esr# clear ip nhrp shortcut-routes network 1.0.0.11/32
```

### clear tunnels counters

Данной командой осуществляется сброс счетчиков заданного туннеля или группы туннелей.

### Синтаксис

```
clear tunnels counters [{ <TUN> | pseudowire [ <ID> <NEIGBOR-ADDR> ] | dypseudowire
pseudowire [ <ID> <NEIGBOR-ADDR> ] } ]
```

### Параметры

<TUN> – имя туннеля устройства, задается в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#).

Можно указать несколько туннелей перечислением через запятую «,» либо указать диапазон туннелей через дефис «-». Если не указывать индексы туннелей, то будут очищены счетчики всех туннелей заданной группы.

<ID> – идентификатор псевдопровода/динамического псевдопровода.

<NEIGBOR-ADDR> – IP-адрес, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример

```
esr# clear tunnels counters dypseudowire 2 192.0.2.10
```

### clear tunnels softgre

Данная команда разрушает softgre-туннели/туннель.

**Синтаксис**

```
clear tunnels softgre [ remote-address <REMOTE-IP> ]
```

**Параметры**

<REMOTE-IP> – удаленный IP-адрес, с которого поднят softgre-туннель.

**Необходимый уровень привилегий**

15

**Командный режим**

ROOT

**Пример**

```
esr# clear tunnels softgre  
esr# clear tunnels softgre remote-address 10.10.42.10
```

**compression**

Данной командой включается механизм сжатия передаваемых данных между клиентами и сервером OpenVPN.

Использование отрицательной формы команды (no) отключает механизм сжатия передаваемых данных.

**Синтаксис**

```
[no] compression
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Выключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-OPENVPN

**Пример**

```
esr(config-openvpn)# compression
```

**default-profile**

Данная команда позволяет использовать конфигурацию данного SoftGRE-туннеля для автоматического создания туннелей с такими же mode и local address.

Использование отрицательной формы команды (no) запрещает использование конфигурации туннеля для автоматического создания туннелей.

**Синтаксис**

```
[no] default-profile
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-SOFTGRE

**Пример**

```
esr(config-softgre)# default-profile
```

**description**

Данная команда используется для изменения описания конфигурируемого туннеля.

Использование отрицательной формы команды (no) удаляет установленное описание.

**Синтаксис**

```
description <DESCRIPTION>
```

```
no description
```

**Параметры**

<DESCRIPTION> – описание туннеля, задаётся строкой до 255 символов.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IP4IP4

CONFIG-SUBTUNNEL

CONFIG-GRE

CONFIG-L2TP  
CONFIG-L2TPV3  
CONFIG-VTI  
CONFIG-LT  
CONFIG-PPTP  
CONFIG-PPPOE  
CONFIG-OPENVPN  
CONFIG-WIREGUARD  
CONFIG-WIREGUARD-TUNNEL-PEER

### Пример

```
esr(config-gre)# description "tunnel to branch"
```

### dscp

Команда задаёт значение кода DSCP для использования в IP-заголовке инкапсулирующего пакета. Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

### Синтаксис

```
dscp <DSCP>  
no dscp
```

### Параметры

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

### Значение по умолчанию

Наследуется от инкапсулируемого пакета.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IP4IP4  
CONFIG-GRE

### Пример

```
esr(config-ip4ip4)# dscp 40
```

**enable**

Данной командой включается туннель.

Использование отрицательной формы команды (no) отключает туннель.

**Синтаксис**

```
[no] enable
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Туннель выключен.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-SUBTUNNEL

CONFIG-L2TP

CONFIG-L2TPV3

CONFIG-VTI

CONFIG-LT

CONFIG-PPTP

CONFIG-PPPOE

CONFIG-OPENVPN

CONFIG-WIREGUARD

CONFIG-WIREGUARD-TUNNEL-PEER

**Пример**

```
esr(config-gre)# enable
```

**encryption algorithm**

Данной командой выбирается алгоритм шифрования, используемый при передачи данных.

Использование отрицательной формы команды (no) отключает шифрование.

**Синтаксис**

```
encryption algorithm <ALGORITHM>  
no encryption algorithm
```

**Параметры**

<ALGORITHM> – идентификатор протокола шифрования, принимает значения: des, blowfish128, aes128, des-ede, aes192, 3des, desx, aes256.

**Значение по умолчанию**

Шифрование отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-OPENVPN

**Пример**

```
esr(config-openvpn)# encryption algorithm aes128
```

**history statistics**

Данной командой включается запись статистики использования текущего туннеля.

Использование отрицательной формы команды (no) отключает запись статистики использования текущего туннеля.

**Синтаксис**

```
[no] history statistics
```

**Параметры**

Отсутствуют.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-GRE

CONFIG-SOFTGRE

CONFIG-SUBTUNNEL

CONFIG-IP4IP4

CONFIG-L2TPV3  
CONFIG-LT  
CONFIG-VTI  
CONFIG-PPTP  
CONFIG-PPPOE  
CONFIG-OPENVPN  
CONFIG-L2TP

### Пример

```
esr(config-ip4ip4)# history statistics
```

### ignore-default-route

Данная команда включает режим, в котором маршрут по умолчанию, полученный от сервера, не устанавливается в таблицу маршрутизации.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
[no] ignore-default-route
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Маршрут по умолчанию, полученный от сервера, устанавливается в таблицу маршрутизации.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-L2TP  
CONFIG-PPTP  
CONFIG-PPPOE

### Пример

```
esr(config-pptp)# ignore-default-route
```

### interface

Данной командой определяется интерфейс, через который будет устанавливаться PPPoE-соединение.

Использование отрицательной формы команды (no) удаляет указанный интерфейс.



### Синтаксис

```
interface <IF>  
no interface
```

### Параметры

<IF> – интерфейс или группы интерфейсов, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-PPPOE

### Пример

```
esr(config-pppoe)# interface gigabitethernet 1/0/5.100
```

### ip dont-fragment-bit ignore

Данной командой включается безусловная фрагментация GRE-трафика. В результате, если размер получившегося после инкапсуляции в GRE пакета больше MTU исходящего интерфейса, то GRE-пакет будет фрагментирован. В фрагментированном GRE-пакете DF-бит будет сброшен.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
[no] ip dont-fragment-bit ignore
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию.

Отключено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GRE

## Пример

```
esr(config-gre)# ip dont-fragment-bit ignore
```

## ip nhrp attribute group

Данная команда задаёт имя NHRP-группы, которое будет передаваться NHRP-соседям в NHRP-сообщениях.

Использование отрицательной формы команды (no) отключает рассылку NHRP-группы.

## Синтаксис

```
ip nhrp attribute group <WORD>
```

```
[no] ip nhrp attribute group
```

## Параметры

<WORD> – имя NHRP-группы, задаётся строкой [1..40] символов, не принимает символы [^#].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-GRE

## Пример

```
esr(config-gre)# ip nhrp attribute group esr-spoke-5mbps
```

## ip nhrp authentication

Данная команда включает аутентификацию для протокола NHRP. У всех участников NHRP-процесса должен быть одинаковый пароль.

Использование отрицательной формы команды (no) отключает аутентификацию.

## Синтаксис

```
ip nhrp authentication { <WORD> | encrypted <ENCRYPTED-TEXT> }
```

```
[no] ip nhrp authentication
```

## Параметры

<WORD> – пароль в открытой форме, задаётся строкой [1..8] символов, может включать символы [0-9a-fA-F];

<ENCRYPTED-TEXT> – зашифрованный пароль, размером [1..8] байт, задаётся строкой [2..16] символов.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-GRE

## Пример

```
esr(config-gre)# ip nhrp authentication password
```

## ip nhrp enable

Данной командой включается работа протокола NHRP на GRE-туннеле маршрутизатора.

Использование отрицательной формы команды (no) отключает работу протокола NHRP на GRE-туннеле маршрутизатора.

## Синтаксис

```
[no] ip nhrp enable
```

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-GRE

## Пример

```
esr(config-gre)# ip nhrp enable
```

## ip nhrp holding-time

Данная команда служит для установки времени жизни NHRP-записи, которая будет сформирована NHRP-сервере после успешной регистрации NHRP-клиента. Аналогичным образом это время будет использоваться в записи на NHRP-клиенте после успешного построения туннеля между NHRP-клиентами. Также эта команда влияет на период повторных запросов регистрации на NHRP-сервере, он рассчитывается как 1/3 от значения команды *ip nhrp holding-time*.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
ip nhrp holding-time <TIME>
```

```
[no] ip nhrp holding-time
```

### Параметры

<TIME> – время в секундах, в течении которого у NHRP-соседа будет существовать запись о данном NHRP-клиенте, принимает значения [1..65535].

### Значение по умолчанию

7200 секунд

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GRE

### Пример

```
esr(config-gre)# ip nhrp holding-time 300
```

### ip nhrp ipsec

Данной командой указывается имя IPsec VPN, который будет использован в качестве транспорта для GRE-туннеля с настроенным протоколом NHRP.

Использование отрицательной формы команды (no) отключает использование IPsec VPN на GRE-туннеле.

### Синтаксис

```
ip nhrp ipsec <WORD> { static | dynamic }  
no ip nhrp ipsec <WORD> { static | dynamic }
```

### Параметры

<WORD> – имя VPN, задаётся строкой до 31 символа;

static – статическое соединение, применяется для связи с NHRP-сервером;

dynamic – динамически устанавливаемое соединение, настраивается для возможности поднятия туннелей до NHRP-клиентов.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-GRE

## Пример

```
esr(config-gre)# ip nhrp ipsec VPN static
```

### ip nhrp map

Данная команда задаёт соответствие «внутреннего» туннельного адреса с «внешним» NBMA-адресом.

Использование отрицательной формы команды (no) удаляет соответствие.

### Синтаксис

```
[no] ip nhrp map <ADDR-IN> <ADDR-OUT>
```

### Параметры

<ADDR-IN> – туннельный IP-адрес задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

<ADDR-OUT> – NBMA IP-адрес задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GRE

## Пример

```
esr(config-gre)# ip nhrp map 172.16.0.2 67.86.141.231
```

### ip nhrp map group

Данная команда задаёт соответствие NHRP-группы, полученной от NHRP-соседа в сообщениях NHRP и политики QoS, которая будет применена к исходящему в сторону этого NHC-соседа трафика.

Использование отрицательной формы команды (no) удаляет соответствие.

### Синтаксис

```
[no] ip nhrp map group <GROUP> service-policy output <POLICY>
```

### Параметры

<GROUP> – имя NHRP-группы, задаётся строкой [1..40] символов, не принимает символы [#];

<POLICY> – имя QoS-политики, задается строкой [1..31] символов.

### Необходимый уровень привилегий

10

**Командный режим**

CONFIG-GRE

**Пример**

```
esr(config-gre)# ip nhrp map group SPOKE-5M service-policy output POLICY-5M
```

**ip nhrp multicast**

Данная команда определяет адресатов для отправки мультикастного трафика.

Использование отрицательной формы команды (no) удаляет адресата.

**Синтаксис**

```
[no] ip nhrp multicast { dynamic | nhs | <ADDR> }
```

**Параметры**

dynamic – отправляет трафик всем NHRP-соседям, которые установили соединение с текущим хостом через процедуру регистрации;

nhs – отправляет трафик на заданные в конфигурации NHRP-сервера, на которых успешно пройдена регистрация;

<ADDR> – отправляет трафик на заданный туннельный IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-GRE

**Пример**

```
esr(config-gre)# ip nhrp multicast nhs
```

**ip nhrp nhs**

Данная команда служит для указания туннельного адреса NHRP-сервера, к которому после успешной регистрации можно отправлять запросы на поиск NHRP-соседей.

Использование отрицательной формы команды (no) удаляет запись о сервере.

**Синтаксис**

```
ip nhrp nhs <ADDR>
```

```
no ip nhrp nhs <ADDR>
```

## Параметры

<ADDR> – туннельный IP-адрес NHRP-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть AAA – DDD принимает значения [0..255].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-GRE

## Пример

```
esr(config-gre)# ip nhrp nhs 172.16.0.2
```

## ip nhrp redirect

Данной командой включается механизм, который позволяет NHRP-серверу отслеживать не оптимальность прохождения трафика между NHRP-соседями. Если NHRP-сервер получает пакет от одного NHRP-соседа и пересылает его другому NHRP-соседу, то отправителю данного пакета будет отправлено NHRP-сообщение Traffic Indication.

Использование отрицательной формы команды (no) отключает данный механизм.

## Синтаксис

```
[no] ip nhrp redirect
```

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-GRE

## Пример

```
esr(config-gre)# ip nhrp redirect
```

## ip nhrp shortcut

Данной командой включается механизм, позволяющий обрабатывать NHRP-сообщения Traffic Indication. Если такое сообщение получено от NHRP-сервера на отправленный ранее пакет, то NHRP-клиент начнет процедуру поиска NHRP-соседа, за которым доступен адрес назначения в отправленном пакете.

Успешный поиск приведет к созданию туннеля между NHRP-соседями для оптимального прохождения трафика.

Использование отрицательной формы команды (no) отключает данный механизм.

### Синтаксис

```
[no] ip nhrp shortcut
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GRE

### Пример

```
esr(config-gre)# ip nhrp shortcut
```

### ip path-mtu-discovery discovery disable

Данной командой отключается функция Path MTU Discovery. В результате, если каждое из следующих условий будет выполнено, то произойдет фрагментация GRE-пакета:

- Размер получившегося после инкапсуляции в GRE пакета больше MTU исходящего интерфейса;
- DF bit (don't fragment bit) исходного пакета не установлен.

В фрагментированном GRE-пакете DF-бит будет унаследован от оригинального пакета.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
[no] ip path-mtu-discovery disable
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию.

Включено.

### Необходимый уровень привилегий

10



## Командный режим

CONFIG-GRE

### Пример

```
esr(config-gre)# ip path-mtu-discovery disable
```

### ip tcp adjust-mss

Данной командой переопределяется значение поля MSS (Maximum segment size) во входящих TCP-пакетах.

Использование отрицательной формы команды (no) отключает корректировку значение поля MSS.

### Синтаксис

```
ip tcp adjust-mss <MSS>
```

```
no ip tcp adjust-mss
```

### Параметры

<MSS> – значение MSS, принимает значения в диапазоне [500..1460].

### Значение по умолчанию

1460

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-L2TP

CONFIG-PPPOE

CONFIG-PPTP

CONFIG-VTI

CONFIG-LT

### Пример

```
esr(config-gre)# ip tcp adjust-mss 1400
```

## ipsec authentication method

Данной командой выбирается метод аутентификации по ключу для IKE-соединения. Аутентификация сообщений по ключу используется при установлении IKE-соединения, ключ задаётся командой «ipsec authentication pre-shared-key» (см. раздел [ipsec authentication pre-shared-key](#)).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ipsec authentication method pre-shared-key
no ipsec authentication method
```

### Параметры

pre-shared-key – метод аутентификации, использующий предварительно полученные ключи шифрования.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-L2TP

### Пример

```
esr(config-l2tp-server)# ipsec authentication method psk
```

## ipsec authentication pre-shared-key

Данной командой устанавливается общий секретный ключ для аутентификации, который должен совпадать у обеих сторон, устанавливающих туннель.

Использование отрицательной формы команды (no) удаляет установленный ключ.

### Синтаксис

```
ipsec authentication pre-shared-key { ascii-text { <TEXT> | encrypted <ENCRYPTED-TEXT> } |
hexadecimal {<HEX> | encrypted <ENCRYPTED-HEX> } }
no ipsec authentication pre-shared-key
```

### Параметры

<TEXT> – строка [1..64] ASCII-символов.

<HEX> – число размером [1..32] байт, задаётся строкой [2..128] символов в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

<ENCRYPTED-TEXT> – зашифрованный пароль размером [1..32] байт, задаётся строкой [2..128] символов;

<ENCRYPTED-HEX> – зашифрованное число размером [2..64] байт, задаётся строкой [2..256] символов.

**Значение по умолчанию**

none

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-L2TP

**Пример**

```
esr(config-l2tp-server)# ipsec authentication pre-shared-key ascii-text password
```

**ipsec ike proposal**

Данной командой для L2TP-клиента назначается шаблон, ограничивающий используемые методы аутентификации и шифрования протокола IKE.

Использование отрицательной формы команды (no) удаляет ограничения на использование методов аутентификации и шифрования протокола IKE.

**Синтаксис**

```
[no] ipsec ike proposal <NAME>
```

**Параметры**

<NAME> – имя ранее созданного профиля протокола IKE, задаётся строкой до 31 символа.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-L2TP

**Пример**

```
esr(config-l2tp)# ipsec ike proposal IKE_PROPOSAL
```

**ipsec proposal**

Данной командой для L2TP-клиента назначается шаблон, ограничивающий используемые методы аутентификации и шифрования протокола IPsec.

Использование отрицательной формы команды (no) удаляет ограничения на использование методов аутентификации и шифрования протокола IPsec.

### Синтаксис

```
[no] ipsec ike proposal <NAME>
```

### Параметры

<NAME> – имя ранее созданного профиля IPsec, задаётся строкой до 31 символа.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-L2TP

### Пример

```
esr(config-l2tp)# ipsec proposal IPSEC_PROPOSAL
```

### keepalive dhcp dependent-interface

Данной командой включается механизм перезапроса IP-адресов по протоколу DHCP на указанных интерфейсах при отключении GRE-туннеля по keepalive. Возможно указать до 8 интерфейсов для каждого GRE-туннеля.

Использование отрицательной формы команды (no) отключает механизм перезапроса IP-адресов по протоколу DHCP.

### Синтаксис

```
keepalive dhcp dependent-interface <IF>  
no keepalive dst-address
```

### Параметры

<IF> – физический или агрегированный интерфейс, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

## Командный режим

### CONFIG-GRE

#### Пример

```
esr(config-gre)# keepalive dhcp dependent-interface gi 1/0/1
```

#### keepalive dhcp link-timeout

Данной командой указывается интервал времени между отключением GRE-туннеля и перезапросом IP-адреса на интерфейсе/интерфейсах, указанных командой *keepalive dhcp dependent-interface* (см. раздел [keepalive dhcp dependent-interface](#)).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### Синтаксис

```
keepalive dhcp link-timeout <SEC>
```

```
no keepalive dhcp link-timeout
```

#### Параметры

<SEC> – интервал в секундах между отключением GRE-туннеля и перезапросом IP-адреса на интерфейсе/интерфейсах указанных командой *keepalive dhcp dependent-interface*, принимает значения [1..32767] секунд.

#### Значение по умолчанию

10

#### Необходимый уровень привилегий

10

## Командный режим

### CONFIG-GRE

#### Пример

```
esr(config-gre)# keepalive dhcp link-timeout 90
```

#### keepalive dst-address

Данной командой настраивается IP-адрес для отправки keepalive-пакетов для проверки работоспособности туннеля. Если указанный IP-адрес недоступен, то туннель меняет оперативное состояние на DOWN. Данный параметр имеет значение только при включенном механизме keepalive (см. раздел [keepalive enable](#)).

Использование отрицательной формы команды (no) отключает данную проверку.

**Синтаксис**

```
keepalive dst-address <ADDR>  
no keepalive dst-address
```

**Параметры**

<ADDR> – IP-адрес для проверки работоспособности GRE-туннеля.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-GRE

**Пример**

```
esr(config-gre)# keepalive dst-address 192.168.1.57
```

**keepalive enable**

Данной командой включается проверка доступности удаленного шлюза туннеля. Если удаленный шлюз туннеля недоступен, то туннель меняет оперативное состояние на DOWN.

Использование отрицательной формы команды (no) отключает данную проверку.

**Синтаксис**

```
[no] keepalive enable
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию.**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-GRE

## Пример

```
esr(config-gre)# keepalive enable
```

## keepalive retries

Данная команда определяет количество попыток проверки доступности удаленного шлюза туннеля. По достижению указанного количества неудачных попыток, туннель будет считаться неактивным.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
keepalive retries <VALUE>
```

```
no keepalive retries
```

## Параметры

<VALUE> – количество попыток, принимает значения в диапазоне [1..255].

## Значение по умолчанию

6

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-GRE

## Пример

```
esr(config-gre)# keepalive retries 8
```

## keepalive timeout

Данной командой регулируется период отправки keepalive-пакетов встречной стороне.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
keepalive timeout <TIME>
```

```
no keepalive timeout
```

## Параметры

<TIME> – время в секундах, принимает значения в диапазоне [1..32767].

**Значение по умолчанию**

10

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-GRE

CONFIG-WIREGUARD-TUNNEL-PEER

**Пример**

```
esr(config-gre)# keepalive timeout 18
```

**keepalive timeout ipsec**

Данной командой задается время, которое отводится на восстановление соединения IPsec VPN, по истечении которого маршрутизатор будет перезагружен. Данный функционал работает только когда маршрутизатор работает в режиме OTT. Режим работы OTT включается при производстве устройства.

При использовании отрицательной формы команды (no) значение приводится к дефолтному значению 180.

**Синтаксис**

```
keepalive timeout ipsec <TIME>
```

```
no keepalive timeout ipsec
```

**Параметры**

<TIME> – время в секундах, принимает значения в диапазоне [30-32767].

**Значение по умолчанию**

180

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-GRE

**Пример**

```
esr(config-gre)# keepalive timeout ipsec 1000
```



## key

Данная команда разрешает передачу ключа (key) в туннельном заголовке GRE (в соответствии с RFC 2890) и устанавливает значение ключа. Ключ может быть использован для идентификации потоков трафика в GRE-туннеле.

Использование отрицательной формы команды (no) запрещает передачу ключа.

### Синтаксис

```
key <KEY>
```

```
no key
```

### Параметры

<KEY> – значение KEY, принимает значения в диапазоне [1..4294967295].

### Значение по умолчанию

Ключ не передаётся.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GRE

### Пример

```
esr(config-gre)# key 40
```

## load-average

Данной командой устанавливается интервал времени, за который усредняется статистика о нагрузке на туннеле.

Использование отрицательной формы команды устанавливает значение по умолчанию.

### Синтаксис

```
load-average <TIME>
```

```
no load-average
```

### Параметры

<TIME> – интервал в секундах, принимает значения [5..150].

### Значение по умолчанию

5

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT

CONFIG-SUBTUNNEL

CONFIG-L2TPv3

CONFIG-VTI

CONFIG-PPTP

CONFIG-PPPOE

CONFIG-OPENVPN

CONFIG-L2TP

**Пример**

```
esr(config-gre)# load-average
```

**local address**

Данной командой устанавливается IP-адрес локального шлюза туннеля.

Использование отрицательной формы команды (no) удаляет IP-адрес локального шлюза.

**Синтаксис**

```
local address <ADDR>
```

```
no local address
```

**Параметры**

<ADDR> – IP-адрес локального шлюза.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-L2TPV3

CONFIG-VTI

## Пример

```
esr(config-ip4ip4)# local address 192.168.1.1
```

### local address xauth

Данной командой задается использование адреса, выдаваемого по mode config, при использовании ранее настроенного IPsec VPN в режиме XAUTH-клиента.

При использовании отрицательной формы команды (no) удаляет настройку.

### Синтаксис

```
local address xauth <NAME>
```

```
no local address
```

### Параметры

<NAME> – имя ранее созданного IPsec VPN, задается строкой до 31 символа.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GRE

## Пример

```
esr(config-)# local address xauth IPsecVPN
```

### local checksum

Данная команда включает вычисление контрольной суммы и занесение её в GRE-заголовок отправляемых пакетов.

Использование отрицательной формы команды (no) отключает процесс вычисления и отправки контрольной суммы.

### Синтаксис

```
[no] local checksum
```

### Параметры

Команда не содержит параметров.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-GRE

**Пример**

```
esr(config-gre)# local checksum
```

**local cookie**

Данная команда определяет значение cookie для дополнительной проверки соответствия между передаваемыми данными и сессией.

Использование отрицательной формы команды (no) удаляет локальный cookie.

**Синтаксис**

```
local cookie <COOKIE>
```

```
no local cookie
```

**Параметры**

<COOKIE> – значение cookie, параметр принимает значения длиной восемь или шестнадцать символов в шестнадцатеричном виде [8 или 16].

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-L2TPV3

**Пример**

```
esr(config-l2tpv3)# local cookie 8FB51B8FB
```

**local interface**

Данной командой устанавливается использование IP-адреса, назначенного на интерфейс в качестве локального шлюза GRE-туннеля.

При использовании отрицательной формы команды (no) прекращается использование IP-адреса, назначенного на интерфейс в качестве локального шлюза.

**Синтаксис**

```
local interface { <IF> | <TUN> }
```

```
no local interface
```

### Параметры

<IF> – тип и идентификатор интерфейса, задаётся в виде, описанном в разделе [Типы и порядок именованя интерфейсов маршрутизатора](#);

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе [Типы и порядок именованя туннелей маршрутизатора](#).

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GRE

### Пример

```
esr(config-gre)# local interface gigabitethernet 1/0/1
```

### local port

Команда определяет локальный UDP-порт, если в качестве метода инкапсуляции был выбран UDP-протокол.

Использование отрицательной формы команды (no) удаляет локальный номер UDP-порта.

### Синтаксис

```
local port <UDP>
```

```
no local port
```

### Параметры

<UDP> – номер UDP-порта в диапазоне [1..65535].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-L2TPV3

### Пример

```
esr(config-l2tpv3)# local port 1501
```

### local session-id

Установить локальный идентификатор сессии.

Использование отрицательной формы команды (no) удаляет локальный идентификатор сессии.

### Синтаксис

```
local session-id <SESSION-ID>
```

```
no local session-id
```

### Параметры

<SESSION-ID> – идентификатор сессии, принимает значения [1..200000].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-L2TPV3

### Пример

```
esr(config-l2tpv3)# local session-id 200
```

### min-frag-size

Данной командой устанавливается минимальный размер пакета (в байтах). В результате пакеты, превышающие заданный размер, будут фрагментироваться при прохождении через соответствующий multilink PPP.

Использование отрицательной формы команды (no) удаляет пороговое значение.

### Синтаксис

```
min-frag-size <SIZE>
```

```
no lmin-frag-size
```

### Параметры

<SIZE> – размер пакета, принимает значения в диапазоне [128..1500].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-MULTILINK

### Пример

```
esr(config-if-multilink)# min-frag-size 1000
```

## mode

Данной командой задается режим работы SoftGRE-туннеля.

Использование отрицательной формы команды (no) снимает установленный режим.

### Синтаксис

```
mode <MODE>  
no mode
```

### Параметры

<MODE> – режим работы туннеля, возможные значения:

- data – режим данных;
- management – режим управления.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-SOFTGRE

### Пример

```
esr(config-softgre)# mode data
```

## mode

Данной командой указывается режим инкапсуляции для GRE-туннеля.

Использование отрицательной формы команды (no) устанавливает инкапсуляцию по умолчанию.

### Синтаксис

```
mode <MODE>
```

### Параметры

<MODE> – режим инкапсуляции для GRE-туннеля:

- ip – инкапсуляция IP-пакетов в GRE;
- ethernet – инкапсуляция Ethernet-фреймов в GRE.

### Значение по умолчанию

ip

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-GRE

### Пример

```
esr(config-gre)# mode ethernet
```

### mtu

Данной командой указывается размер MTU (Maximum Transmission Unit) для туннелей.

Использование отрицательной формы команды (no) устанавливает значение MTU по умолчанию.

### Синтаксис

```
mtu <MTU>
```

```
no mtu
```

### Параметры

<MTU> – значение MTU, принимает значения в диапазоне [552..10000] (для PPPoE-туннелей принимает значения в диапазоне [552..1500]).

### Значение по умолчанию

1500

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-SUBTUNNEL

CONFIG-L2TP

CONFIG-L2TPV3

CONFIG-VTI

CONFIG-LT

CONFIG-PPTP

CONFIG-PPPOE

CONFIG-OPENVPN

CONFIG-WIREGUARD



## Пример

```
esr(config-l2tpv3)# mtu 1400
```

## multipoint

Данная команда служит для перевода туннеля в режим multipoint. В этом режиме возможно установление нескольких соединений с одного туннельного интерфейса.

Использование отрицательной формы команды (no) переводит в обычный, point-to-point режим.

## Синтаксис

```
[no] multipoint
```

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-GRE

## Пример

```
esr(config-gre)# multipoint
```

## password

Команда для установки пароля пользователя OpenVPN-сервера.

Использование отрицательной формы команды (no) удаляет пароль пользователя.

## Синтаксис

```
password { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
```

```
no password
```

## Параметры

<CLEAR-TEXT> – пароль в открытой форме, задаётся строкой [8..32] символов, может включать символы [0-9a-fA-F];

<ENCRYPTED-TEXT> – пароль в зашифрованной форме, задаётся строкой [8..32] символов.

- ✓ Пароли хранятся в конфигурации в зашифрованной форме независимо от формата, использованного при вводе команды.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-OPENVPN

## Пример

```
esr(config-openvpn)# password 01234567
```

## peer

Данная команда используется для перехода к настройке разрешённых туннелей. Использование отрицательной формы команды (no) удаляет разрешённый туннель.

## Синтаксис

```
[no] peer <COUNT>
```

## Параметры

<COUNT> – номер соответствующего пира, принимает значения [1..16].

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-WIREGUARD

## Пример

```
esr(config-wireguard)# peer 1
```

## peer lt

Данная команда используется для задания удаленной стороны (в другом VRF) логического туннеля. Использование отрицательной формы команды (no) удаляет привязку удаленной стороны туннеля.

## Синтаксис

```
[no] peer lt <ID>
```

## Параметры

<ID> – идентификатор удаленной стороны логического туннеля.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-LT

**Пример**

```
esr(config-lt)# peer lt 2
```

**port**

Данной командой определяется номер UDP-порта, по которому устанавливается соединение с L2TP-сервером.

Использование отрицательной формы команды (no) удаляет указанный метод аутентификации.

**Синтаксис**

```
port <PORT>
```

```
no port
```

**Параметры**

<PORT> – номер UDP-порта, задаётся в диапазоне [1024..65535].

**Значение по умолчанию**

1701

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-L2TP

**Пример**

```
esr(config-l2tp)# port 1048
```

**port**

Данной командой определяется номер UDP-порта, по которому устанавливается соединение с WireGuard-сервером.

Использование отрицательной формы команды (no) удаляет указанный метод аутентификации.

**Синтаксис**

```
port <PORT>  
no port
```

**Параметры**

<PORT> – номер UDP-порта, задаётся в диапазоне [1..65535].

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WIREGUARD

**Пример**

```
esr(config-wireguard)# port 43020
```

**ppp failure-count**

Данной командой устанавливается количество неудачных data-link тестов перед разрывом сессии. При использовании отрицательной формы команды (no) устанавливается значение по умолчанию.

**Синтаксис**

```
ppp failure-count <NUM>  
no ppp failure-count
```

**Параметры**

<NUM> – количество неудачных data-link тестов, задается в диапазоне [1..100].

**Значение по умолчанию**

10

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-L2TP  
CONFIG-PPPOE  
CONFIG-PPTP

## Пример

```
esr(config-l2tp)# ppp failure-count 20
```

## ppp timeout keepalive

Данной командой устанавливается интервал времени в секундах, по истечении которого маршрутизатор отправляет keepalive-сообщение.

При использовании отрицательной формы команды (no) устанавливается значение по умолчанию.

## Синтаксис

```
ppp timeout keepalive <TIME >
```

```
no ppp timeout keepalive
```

## Параметры

<TIME> – время в секундах, задается в диапазоне [1..32767].

## Значение по умолчанию

10

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-L2TP

CONFIG-PPPOE

CONFIG-PPTP

## Пример

```
esr(config-l2tp)# ppp timeout keepalive 5000
```

## protocol

Выбор метода инкапсуляции для туннеля L2TPv3.

## Синтаксис

```
protocol <TYPE>
```

```
no protocol
```

## Параметры

<TYPE> – тип инкапсуляции, возможные значения:

- IP-инкапсуляция в IP-пакет;
- UDP-инкапсуляция в UDP-дейтаграммы.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-L2TPV3

### Пример

```
esr(config-l2tpv3)# protocol ip
```

### private-key

Данной командой указывается необходимый приватный ключ клиента WireGuard.

### Синтаксис

```
private-key <NAME>  
no private-key
```

### Параметры

<NAME> – имя приватного ключа, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-WIREGUARD

### Пример

```
esr(config-wireguard)# private-key client.priv
```

### protocol

Выбор метода инкапсуляции для туннеля OPENVPN.

### Синтаксис

```
protocol <TYPE>  
no protocol
```

## Параметры

<TYPE> – тип инкапсуляции, возможные значения:

- TCP – инкапсуляция в TCP-сегмент;
- UDP – инкапсуляция в UDP-дейтаграммы.

## Значение по умолчанию

TCP

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-OPENVPN

## Пример

```
esr(config-openvpn)# protocol tcp
```

## public-key

Данной командой указывается необходимый публичный ключ сервера WireGuard.

## Синтаксис

public-key <NAME>

no public-key

## Параметры

<NAME> – имя публичного ключа, задаётся строкой до 31 символа.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-WIREGUARD-TUNNEL-PEER

## Пример

```
esr(config-wireguard-tunnel-peer)# public-key server.pub
```

## remote address

Данной командой устанавливается IP-адрес удаленного шлюза туннеля.

Использование отрицательной формы команды (no) удаляет IP-адрес удаленного шлюза.

### Синтаксис

```
remote address <ADDR>
```

```
no remote address
```

### Параметры

<ADDR> – IP-адрес удаленного шлюза.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-SOFTGRE

CONFIG-L2TP

CONFIG-L2TPV3

CONFIG-VTI

CONFIG-PPTP

CONFIG-WIREGUARD-PEER

### Пример

```
esr(config-ip4ip4)# remote address 192.168.1.2
```

### remote address

Данной командой устанавливается IP-адрес и TCP/UDP-порт удаленного шлюза туннеля.

Использование отрицательной формы команды (no) удаляет IP-адрес удаленного шлюза.

### Синтаксис

```
[no] remote address <ADDR> [ port <PORT>]
```

### Параметры

<ADDR> – IP-адрес удаленного шлюза;

<PORT> – номер TCP/UDP-порта удаленного шлюза в диапазоне [1..65535].

### Значение по умолчанию

<PORT> – 1194.



**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-OPENVPN

**Пример**

```
esr(config-openvpn)# remote address 192.168.1.2 port 1233
```

**remote address xauth**

Данной командой задается использование адреса management-ip или data-ip, выдаваемого по mode config, при использовании ранее настроенного IPsec VPN в режиме XAUTH-клиента. Требуется наличия соответствующих настроек на IPsec-VPN-сервере.

При использовании отрицательной формы команды (no) удаляет настройку.

**Синтаксис**

```
remote address xauth <NAME> {management-ip|data-ip}  
no remote address
```

**Параметры**

<NAME> – имя ранее созданного IPsec VPN, задается строкой до 31 символа;

management-ip – адрес, получаемый по mode config при установке IPsec VPN в режиме XAUTH-клиента. Требуется наличия в настройках IPsec-VPN-сервера роутера ELTEX\_MANAGEMENT\_IP (28683).

data-ip – адрес, получаемый по mode config при установке IPsec VPN в режиме XAUTH-клиента. Требуется наличия в настройках IPsec-VPN-сервера ELTEX\_DATA\_IP (28684).

**Значение по умолчанию**

Не задано.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-GRE

**Пример**

```
esr(config-gre)# remote address xauth IPsecVPN
```

## remote checksum

Команда включает проверку наличия и соответствия значений контрольной суммы в заголовках принимаемых GRE-пакетов.

Использование отрицательной формы команды (no) отключает проверку контрольной суммы.

### Синтаксис

```
[no] remote checksum
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

По умолчанию проверка контрольной суммы выключена.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GRE

### Пример

```
esr(config-gre)# remote checksum
```

## remote cookie

Данная команда определяет значение cookie для дополнительной проверки соответствия между передаваемыми данными и сессией.

Использование отрицательной формы команды (no) удаляет удаленный cookie.

### Синтаксис

```
remote cookie <COOKIE>
```

```
no remote cookie
```

### Параметры

<COOKIE> – значение cookie, принимает значения длиной восемь или шестнадцать символов в шестнадцатеричном виде [8 или 16].

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-L2TPV3

### Пример

```
esr(config-l2tpv3)# remote cookie 8FB51B8FB
```

### remote port

Данная команда определяет удаленный UDP-порт, если в качестве метода инкапсуляции был выбран UDP.

Использование отрицательной формы команды (no) удаляет удаленный номер UDP-порта.

### Синтаксис

```
remote port <UDP>
```

```
no remote port
```

### Параметры

<UDP> – номер UDP-порта в диапазоне [1..65535].

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-L2TPV3

CONFIG-WIREGUARD-PEER

### Пример

```
esr(config-l2tpv3)# remote port 65000
```

### remote session-id

Данной командой устанавливается удаленный идентификатор сессии.

Использование отрицательной формы команды (no) удаляет удаленный идентификатор сессии.

### Синтаксис

```
remote session-id <SESSION-ID>
```

```
no remote session-id
```

## Параметры

<SESSION-ID> – идентификатор сессии, принимает значение в диапазоне [1..200000].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-L2TPV3

## Пример

```
esr(config-l2tpv3)# remote session-id 2
```

## route-metric

Данной командой назначается метрика маршрутов, получаемых клиентом от OpenVPN-сервера. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
route-metric <METRIC>  
no route-metric
```

## Параметры

<METRIC> – метрика маршрута, принимает значение в диапазоне [0..255].

## Значение по умолчанию

0

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-OPENVPN

## Пример

```
esr(config)# route-metric 100
```

## route-nopull

Данной командой отключается применение маршрутов, передаваемых OpenVPN-сервером. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
[no] route-nopull
```

**Параметры**

Отсутствуют.

**Значение по умолчанию**

Игнорирование маршрутов отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-OPENVPN

**Пример**

```
esr(config)# route-nopull
```

**snmp init-trap**

Данная команда используется для включения отправки snmp-trap о включении/отключении туннеля.

Использование отрицательной формы команды (no) отключает отправки snmp-trap о включении/отключении туннеля.

**Синтаксис**

```
[no] snmp init-trap
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-GRE

CONFIG-SUBTUNNEL

## Пример

```
esr(config-gre)# snmp init-trap
```

## show ip nhrp peers

Данная команда служит для просмотра записей об NHRP-соседях.

### Синтаксис

```
show ip nhrp peers [ { connected | group [ name <GROUP> ] | lower-up | nat | nhs |
protected | qos | unique | used } ] [ { nbma-address <ADDR> | tunnel gre <ID> | tunnel-
address <ADDR> } ] [ type { cached | dynamic | incomplete | negative | static } ] [ vrf
<VRF> ] [ detailed ]
```

### Параметры

**connected** – показать записи, для которых разрешена отправка пользовательских пакетов через GRE-туннель;

**group** – показать записи, в которых NHRP-сосед сообщил свою NHRP-группу, при опциональном указании имени NHRP-группы будут выведены NHRP-записи с принадлежностью к указанной группе;

**<GROUP>** – имя NHRP-группы, задаётся строкой [1..40] символов, не принимает символы [^#];

**lower-up** – показать записи, в которых оперативное состояние GRE-туннеля – UP;

**nat** – показать записи, в которых NHRP-сосед расположен за NAT;

**nhs** – показать записи, в которых осуществляется регистрация на NHRP-сервере;

**protected** – показать записи, в которых GRE-туннель до NHRP-соседа защищен шифрованием IPsec;

**qos** – показать записи, в которых к GRE-туннелю до NHRP-соседа применена QoS политика на основании NHRP-группы;

**unique** – показать записи, для которых требуется уникальность соответствия туннельного и NBMA-адреса;

**used** – показать записи, в которых через GRE-туннель проходит пользовательский трафик;

**<ADDR>** – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

**<ID>** – идентификатор GRE-туннеля;

**cached** – показать записи, соответствующие GRE-туннелям между NHRP-клиентами;

**dynamic** – показать записи, соответствующие GRE-туннелям от NHRP-клиента до NHRP-сервера;

**incomplete** – показать незавершенные записи, в которых известен туннельный адрес, но не определен NBMA-адрес;

**negative** – показать ошибочные записи, для которых ни один доступный NHRP-сервер не имеет записи о запрошенном туннельном адресе;

**static** – показать записи, заданные через конфигурацию маршрутизатора;

**<VRF>** – имя экземпляра VRF, задается строкой от 1 до 31 символа;

**detailed** – показать информацию об отфильтрованных NHRP-записях в подробном виде.

### Необходимый уровень привилегий

10

## Командный режим

ROOT

### Пример

```

esr# show ip nhrp peers
Flags: E - unique, R - nhs, U - used, L - lower-up
      C - connected, G - group, Q - qos, N - nat
      P - protected, X - undefined

Tunnel address      NBMA address      Tunnel      Expire      Created      Type
Flags
-----
-----
-----
(h:m:s)          (d,h:m:s)
-----
-----
172.16.0.1          10.0.104.2        gre 1       --          --          static
RLC
172.16.0.11        98.0.101.2        gre 1       01:31:56   00,00:28:03    cached
LCN
esr# show ip nhrp peers detailed
Tunnel:                gre 1
Type:                   static
Tunnel address:        172.16.0.1
NBMA address:          10.0.104.2
NAT-OA address:        --
Flags:                  nhs, lower-up, connected
Created (d,h:m:s):     --
Expire (h:m:s):        --
Re-registration in (h:m:s): 00:10:56
IPSec protection:      Disabled
Group:                  --
QoS policy output:     --

Tunnel:                gre 1
Type:                   cached
Tunnel address:        172.16.0.11
NBMA address:          98.0.101.2
NAT-OA address:        10.0.101.2
Flags:                  lower-up, connected, nat
Created (d,h:m:s):     00,00:28:05
Expire (h:m:s):        01:31:54
Re-registration in (h:m:s): --
IPSec protection:      Disabled
Group:                  --
QoS policy output:     --
esr#

```

### show ip nhrp shortcut-routes

Данная команда служит для просмотра созданных shortcut-route записей до подсетей, находящихся за NHRP NHC.

### Синтаксис

```
show ip nhrp shortcut-routes [ { network <ADDR/LEN> | nexthop <ADDR> | tunnel gre <ID> } ] [ vrf <VRF> ]
```

## Параметры

<ADDR/LEN> – IP-подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32];

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<ID> – идентификатор GRE-туннеля;

<VRF> – имя экземпляра VRF, задается строкой от 1 до 31 символа.

## Необходимый уровень привилегий

10

## Командный режим

ROOT

## Пример

```
esr-200# show ip nhrp shortcut-routes
Network                Nexthop                Tunnel                Expire                Created
-----                -
1.0.0.11/32            172.16.0.11            gre 1                 01:59:35             0,00:00:24
1.0.0.12/32            172.16.0.12            gre 1                 01:59:49             0,00:00:10
esr-200#
```

## show tunnels configuration

Командой выполняется просмотр конфигурации туннеля.

## Синтаксис

```
show tunnels configuration [{ <TUN> | pseudowire [ <ID> <NEIGBOR-ADDR> ] | dypseudowire
pseudowire [ <ID> <NEIGBOR-ADDR> ] } ]
```

## Параметры

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#).

<ID> – идентификатор псевдопровода/динамического псевдопровода.

<NEIGBOR-ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

## Необходимый уровень привилегий

1

## Командный режим

ROOT



## Пример

```

esr# show tunnels configuration gre 25
State:                               enabled
Description:
Local address:                       14.0.0.2
Remote address:                      14.0.0.1
Calculates checksums for outgoing GRE packets: no
Requires that all input GRE packets were checksum: no
key:                                  -
TTL:                                  Inherit
DSCP:                                  0
MTU:                                  1500
Security zone:                       remote

```

## show tunnels counters

Командой выполняется просмотр счетчиков на туннелях.

## Синтаксис

```

show tunnels counters [{ <TUN> | pseudowire [ <ID> <NEIGBOR-ADDR> ] | dypseudowire
pseudowire [ <ID> <NEIGBOR-ADDR> ] } ]

```

## Параметры

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#).

Можно указать несколько туннелей. Если не указывать индексы туннелей, то будут отображены счетчики всех туннелей заданной группы. Если задан определённый туннель, то будет отображена детальная информация по данному туннелю.

<ID> – идентификатор псевдопровода/динамического псевдопровода.

<NEIGBOR-ADDR> – IP-адрес задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```

esr# show tunnels counters dypseudowire 2 192.0.2.10
Tunnel 'dypseudowire 2_192.0.2.10' counters:
Packets received:                   1006
Bytes received:                     84200
Dropped on receive:                 0
Receive errors:                     0
Multicasts received:               0
Receive length errors:              0

```

```

Receive buffer overflow errors: 0
Receive CRC errors: 0
Receive frame errors: 0
Receive FIFO errors: 0
Receive missed errors: 0
Receive compressed: 0
Packets transmitted: 1006
Bytes transmitted: 102308
Dropped on transmit: 0
Transmit errors: 0
Transmit aborted errors: 0
Transmit carrier errors: 0
Transmit FIFO errors: 0
Transmit heartbeat errors: 0
Transmit window errors: 0
Transmit compressed: 0
Collisions: 0

```

## show tunnels history

Команда используется для просмотра статистики использования туннеля.

### Синтаксис

```
show tunnels history [{ <TUN> | pseudowire [ <ID> <NEIGBOR-ADDR> ] | dypseudowire
pseudowire [ <ID> <NEIGBOR-ADDR> ] } ] [ timer <TIMER> ]
```

### Параметры

<TUN> – имя туннеля, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#).

<ID> – идентификатор псевдопровода/динамического псевдопровода.

<NEIGBOR-ADDR> – IP-адрес задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<TIMER> – необязательный ключ timer. В качестве параметров для данного ключа могут выступать:

- hours – отображает историю за последние 72 часа;
- minutes – отображает историю за последние 60 минут;
- seconds – отображает историю за последние 60 секунд;
- При отсутствии ключа timer выводятся 3 таблицы истории использования туннеля/туннелей.

### Необходимый уровень привилегий

5

### Командный режим

ROOT

### Пример

```

esr# show tunnel history gre 1 timer minutes
gre 1
Last 60 minutes:

```

Timer	Recv utilization, Kbit/s	Sent utilization, Kbit/s	Recv errors	Sent errors	Output drops
0-1	240	16	0	0	0
1-2	961	64	0	0	0
2-3	962	64	0	0	0
3-4	962	64	0	0	0
4-5	960	64	0	0	0
5-6	961	64	0	0	0
6-7	719	64	0	0	0
7-8	960	64	0	0	0
8-9	800	65	0	0	0
9-10	962	64	0	0	0
10-11	865	64	0	0	0
11-12	962	64	0	0	0
12-13	817	65	0	0	0
13-14	962	65	0	0	0
14-15	961	65	0	0	0
15-16	880	60	0	0	0
16-17	960	63	0	0	0
17-18	0	0	0	0	0
18-19	0	0	0	0	0
19-20	0	0	0	0	0
20-21	0	0	0	0	0
21-22	0	0	0	0	0

## show tunnels status

Команда используется для просмотра состояния системных интерфейсов.

### Синтаксис

```
show tunnels status [{ <TUN> | pseudowire [ <ID> <NEIGBOR-ADDR> ] | dypseudowire
pseudowire [ <ID> <NEIGBOR-ADDR> ] } ]
```

### Параметры

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#);

В команде можно указать несколько туннелей. Если не указывать индексы туннелей, то будут отображены статусы всех туннелей заданной группы. Если задан конкретный туннель, то будет отображена детальная информация по данному туннелю;

<ID> – идентификатор псевдопровода/динамического псевдопровода;

<NEIGBOR-ADDR> – IP-адрес задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

### Необходимый уровень привилегий

1

### Командный режим

ROOT

## Пример

```

esr-12vf# show tunnels status
Tunnel          Admin  Link  MTU    Local IP          Remote IP          Last change
-----  -----  -----  -----  -----  -----  -----
vti 1          Up     Up    1500   3.3.3.2          3.3.3.1           00,00:10:01
gre 1          Up     Up    1500   3.3.3.2          3.3.3.1           00,00:10:01

```

## show tunnels utilization

Команда используется для просмотра средней нагрузки в туннелях за указанный период.

## Синтаксис

```

show tunnels utilization [{ <TUN> | pseudowire [ <ID> <NEIGBOR-ADDR> ] | dypseudowire
pseudowire [ <ID> <NEIGBOR-ADDR> ] } ]

```

## Параметры

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#);

Можно указать несколько туннелей перечислением через «,» либо указать диапазон интерфейсов через «-». Если не указывать индексы туннелей, то будут очищены счетчики всех туннелей заданной группы;

<ID> – идентификатор псевдопровода/динамического псевдопровода;

<NEIGBOR-ADDR> – IP-адрес задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

## Необходимый уровень привилегий

10

## Командный режим

ROOT

## Пример

```

esr# show tunnels utilization gre 2
Tunnel  Period, s  Sent,  Recv,  Frames Sent  Frames Recv
-----  -----  -----  -----  -----  -----
gre 2    15         0       0       0           0

```

## subnet

Команда указывает список IP-адресов, которым будет разрешено находиться внутри туннеля.

## Синтаксис

```

subnet <OBJ-GROUP-NETWORK-NAME>

```

```
no subnet
```

### Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, содержащего префиксы подсетей назначения, задается строкой до 31 символа.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-WIREGUARD-TUNNEL-PEER

### Пример

```
esr(config-wireguard-tunnel-peer)# subnet client
```

### ttl

Команда задаёт значение времени жизни TTL для туннельных пакетов.

Использование отрицательной формы команды (no) устанавливает значение TTL по умолчанию.

### Синтаксис

```
ttl <TTL>
```

```
no ttl
```

### Параметры

<TTL> – значение TTL, принимает значения в диапазоне [1..255].

### Значение по умолчанию

Наследуется от инкапсулируемого пакета.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-IP4IP4

CONFIG-GRE

## Пример

```
esr(config-ip4ip4)# ttl 10
```

## tunnel

Данная команда позволяет перейти в режим конфигурирования туннеля.

Использование отрицательной формы команды (no) удаляет туннель.

## Синтаксис

```
[no] tunnel <TUN>
```

## Параметры

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#).

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример 1

Переход в режим конфигурирования туннеля l2tp 10:

```
esr(config)# tunnel l2tp 10
esr(config-l2tp)#
```

## Пример 2

Переход в режим конфигурирования туннеля l2tpv3 10:

```
esr(config)# tunnel l2tpv3 10
esr(config-l2tpv3)#
```

**Пример 3**

Переход в режим конфигурирования туннеля ip4ip4 200:

```
esr(config)# tunnel ip4ip4 200
esr(config-ip4ip4)#
```

**Пример 4**

Переход в режим конфигурирования туннеля gre 25:

```
esr(config)# tunnel gre 25
esr(config-gre)#
```

**Пример 5**

Переход в режим конфигурирования туннеля vti 125:

```
esr(config)# tunnel vti 125
esr(config-vti)#
```

**Пример 6**

Переход в режим конфигурирования туннеля pptp 10:

```
esr(config)# tunnel pptp 10
esr(config-pptp)#
```

**Пример 7**

Переход в режим конфигурирования туннеля pppoe 8:

```
esr(config)# tunnel pppoe 8
esr(config-pppoe)#
```

**tunnel**

Данной командой указывается режим инкапсуляции для OpenVPN-клиента.

Использование отрицательной формы команды (no) устанавливает инкапсуляцию по умолчанию.

**Синтаксис**

```
tunnel <MODE>
```

**Параметры**

<MODE> – режим инкапсуляции для OpenVPN-клиента:

- ip – инкапсуляция IP-пакетов в OpenVPN;
- ethernet – инкапсуляция Ethernet-фреймов в OpenVPN.

**Значение по умолчанию**

ip

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-OPENVPN

**Пример**

```
esr(config-openvpn)# mode ethernet
```

**tunnel-source**

Данной командой указывается имя VRF, от IP-интерфейса которого будет строиться данный GRE-туннель. Данная команда актуальна в случае если GRE-туннель строится через VRF, отличный от VRF самого туннеля.

Использование отрицательной формы команды (no) устанавливает режим, когда GRE-туннель и IP-интерфейс, от которого строится GRE-туннель, находятся в одном VRF.

**Синтаксис**

```
tunnel-source [ vrf <VRF> ]  
[no] tunnel-source
```

**Параметры**

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

Без указания ключа "vrf" и имени экземпляра VRF, будет использоваться IP-интерфейс глобальной конфигурации.

**Значение по умолчанию**

Отключено (GRE-туннель и IP-интерфейс относятся к одному VRF).

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-GRE

**Пример**

```
esr(config-gre)# tunnel-source vrf magistral
```



**username**

Данной командой задается пользователь и пароль для подключения к L2TP-, PPPoE- или PPTP-серверу. Использование отрицательной формы команды (no) удаляет указанного пользователя.

**Синтаксис**

```
username <NAME> password ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no username <NAME>
```

**Параметры**

<NAME> – имя пользователя, задается строкой до 31 символа;

<CLEAR-TEXT> – пароль, задается строкой [1..64] символов;

<ENCRYPTED-TEXT> – зашифрованный пароль, задается строкой [2..128] символов.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-L2TP

CONFIG-PPPOE

CONFIG-PPTP

**Пример**

```
esr(config-pptp)# username fedor password ascii-text password
```

**username**

Данной командой создается пользователь для подключения к OpenVPN-серверу.

Использование отрицательной формы команды (no) удаляет указанного пользователя.

**Синтаксис**

```
username <NAME>
no username <NAME>
```

**Параметры**

<NAME> – имя пользователя, задается строкой до 31 символа.

**Необходимый уровень привилегий**

15

## Командный режим

### CONFIG-OPENVPN

#### Пример

```
esr(config-openvpn)# username fedor
```

## Настройки IPsec VPN

- [Управление VPN. Настройки IKE](#)
  - [access profile](#)
  - [access-profile](#)
  - [address-assignment pool](#)
  - [assign-interface](#)
  - [authentication algorithm](#)
  - [authentication mode](#)
  - [authentication method](#)
  - [bind-interface vti](#)
  - [crypto](#)
  - [data-tunnel address](#)
  - [dead-peer-detection action](#)
  - [dead-peer-detection interval](#)
  - [dead-peer-detection retransmit jitter](#)
  - [dead-peer-detection retransmit limit](#)
  - [dead-peer-detection retransmit timeout](#)
  - [dead-peer-detection retransmit tries](#)
  - [dead-peer-detection timeout](#)
  - [description](#)
  - [dh-group](#)
  - [encryption algorithm](#)
  - [identity](#)
  - [ike-policy](#)
  - [ip prefix](#)
  - [keyring](#)
  - [lifetime seconds](#)
  - [local address](#)
  - [local id](#)
  - [local interface](#)
  - [local network](#)
  - [management-tunnel address](#)
  - [mobike disable](#)
  - [mode](#)
  - [mode](#)
  - [password](#)
  - [password local-crt-key](#)
  - [pfs dh-group](#)
  - [pre-shared-key](#)
  - [proposal](#)
  - [remote address](#)
  - [remote id](#)
  - [remote network](#)
  - [remote network dynamic client](#)
  - [security ike gateway](#)

- security ike keyring
- security ike policy
- security ike proposal
- security ike session reauthentication
- security ike session uniqueids
- show security ike
- user
- version
- Управление VPN. Настройки IPsec
  - authentication algorithm
  - clear security ipsec vpn
  - description
  - enable
  - encryption algorithm
  - ike dscp
  - ike establish-tunnel
  - ike gateway
  - ike idle-time
  - ike rekey disable
  - ike rekey margin
  - ike rekey randomization
  - ike ipsec-policy
  - lifetime
  - manual authentication algorithm
  - manual authentication key
  - manual bind-interface vti
  - manual encryption algorithm
  - manual encryption key
  - manual mode
  - manual protocol
  - manual spi
  - mode
  - proposal
  - protocol
  - security ipsec policy
  - security ipsec proposal
  - security ipsec vpn
  - show security ipsec
  - show security ipsec vpn authentication
  - show security ipsec vpn status
  - type

## Управление VPN. Настройки IKE

### access profile

Данной командой создается профиль настроек пользователя для IKE-GATEWAY с определенным именем и осуществляется переход в режим конфигурирования профиля.

Использование отрицательной формы команды (no) удаляет сконфигурированный профиль настроек пользователя для IKE-GATEWAY.

### Синтаксис

```
[no] access profile <NAME>
```

**Параметры**

<NAME> – имя профиля пользователя для IKE-GATEWAY, задаётся строкой до 31 символа.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# access profile OFFICE
```

**access-profile**

Данной командой указывается локальный список пользователей для авторизации. В случае настройки на IPsec VPN Remote Access сервере указывается только название локального списка пользователей, а в случае настройки IPsec VPN Remote Access клиента дополнительно указывается имя пользователя, который будет использован для авторизации на сервере.

Использование отрицательной формы команды (no) удаляет заданный профиль.

**Синтаксис**

```
access-profile <NAME> [ client <USER-NAME> ]
[no] access-profile <NAME>
```

**Параметры**

<NAME> – название локального списка пользователей, задаётся строкой до 31 символа;

<USER-NAME> – имя пользователя из прикрепленного, задаётся строкой до 31 символа.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IKE-GATEWAY

**Пример**

```
esr(config-ike-gateway)# access-profile OFFICE client admin
```

**address-assignment pool**

Команда используется для создания пула адресов и настройки передаваемых параметров для динамической конфигурации IPsec-клиентов.

Использование отрицательной формы команды (no) удаляет пул адресов.

**Синтаксис**

```
[no] address-assignment pool <NAME>
```

**Параметры**

<NAME> – имя пула адресов, задаётся строкой до 31 символа.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# address-assignment pool CENTER
esr(config-pool)#
```

**assign-interface**

Данной командой указывается loopback-интерфейс для назначения динамического адреса, получаемого от IPsec-VPN-сервера.

При использовании отрицательной формы команды (no) удаляется loopback-интерфейс для назначения динамического адреса, получаемого от IPsec-VPN-сервера.

**Синтаксис**

```
assign-interface loopback <LOOPBACK> | <LOOPBACK>-<LOOPBACK>
```

```
no assign-interface
```

**Параметры**

<LOOPBACK> – номер созданного ранее loopback-интерфейса, принимает значение в диапазоне [1..8].

**Значение по умолчанию**

Отсутствует

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IKE-GW

**Пример**

```
esr(config-ike-gw)# assign-interface loopback 3
```

## authentication algorithm

Данной командой устанавливается алгоритм аутентификации, который используется для аутентификации сообщений установленного IKE-соединения. При установлении IKE-соединения используется аутентификация сообщений по ключу (authentication, см. [password](#)).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
authentication algorithm <ALGORITHM>
no authentication algorithm
```

### Параметры

<ALGORITHM> – алгоритм аутентификации, принимает значения: md5, sha1, sha2-256, sha2-384, sha2-512.

### Значение по умолчанию

sha1

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IKE-PROPOSAL

### Пример

```
esr(config-ike-proposal)# authentication algorithm md5
```

## authentication mode

Данной командой устанавливается режим XAUTH-аутентификации удаленных пользователей, подключающихся по протоколу IPsec.

Использование отрицательной формы команды (no) удаляет установленный режим.

### Синтаксис

```
authentication mode { local | radius | client }
no authentication mode
```

### Параметры

local – режим аутентификации, использующий локальную базу пользователей конфигурируемого профиля;

radius – режим, при котором аутентификация пользователей проходит через RADIUS-сервер;

client – режим, используемый XAUTH-клиентом.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IKE-POLICY

## Пример

```
esr(config-ike-policy)# authentication mode local
```

## authentication method

Данной командой выбирается метод аутентификации по ключу для IKE-соединения. Аутентификация сообщений по ключу используется при установлении IKE-соединения, ключ задаётся в IKE-политике (см. раздел [pre-shared-key](#)). После установления IKE-соединения аутентификация сообщений осуществляется при помощи алгоритма хеширования.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
authentication method <METHOD>
```

```
no authentication method
```

## Параметры

<METHOD> – метод аутентификации ключа. Может принимать значения:

- pre-shared-key – метод аутентификации, использующий предварительно полученные ключи шифрования;
- public-key – метод аутентификации, использующий публичный ключ сертификата;
- xauth-psk-key – метод расширенной аутентификации, использующий локальные или удаленные базы пользователей;
- eap – метод расширенной аутентификации, использующий пару логин-пароль и предварительно полученные сертификаты;
- keyring – метод аутентификации, использующий набор ключей аутентификации.

## Значение по умолчанию

pre-shared-key

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IKE-POLICY

## Пример

```
esr(config-ike-proposal)# authentication method pre-shared-key
```

**bind-interface vti**

Данной командой указывается туннельный интерфейс, через который будет проходить трафик в режиме туннеля «route-based».

Использование отрицательной формы команды (no) удаляет привязку к туннельному интерфейсу.

**Синтаксис**

```
bind-interface vti <VTI>
no bind-interface vti
```

**Параметры**

<VTI> – идентификационный номер интерфейса VTI.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IKE-GATEWAY

**Пример**

```
esr(config-ike-gw)# bind-interface vti 1
```

**crypto**

Данной командой указываются необходимые сертификаты.

Использование отрицательной формы команды (no) удаляет название сертификата из конфигурации.

**Синтаксис**

```
crypto <CERTIFICATE-TYPE> <NAME>
no crypto <CERTIFICATE-TYPE>
```

**Параметры**

<CERTIFICATE-TYPE> – тип сертификата или ключа, может принимать следующие значения:

- ca – сертификат центра сертификации;
- crl – список отозванных сертификатов;
- local-crt – сертификат локальной стороны;
- local-crt-key – ключ сертификата локальной стороны;
- remote-crt – сертификат удаленной стороны. Вместо имени файла возможно использование ключа "any" для установления режима приема открытого ключа удаленной стороны по сети;

<NAME> – имя сертификата или ключа, задается строкой до 31 символа.

**Значение по умолчанию**

Отсутствует



**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IKE-POLICY

**Пример**

```
esr(config-ike-policy)# crypto ca KEY
```

**data-tunnel address**

Данной командой указывается IP-адрес для постройки GRE data туннеля, передаваемого клиенту, подключаемому через IPsec с использованием динамического конфигурирования параметров. GRE data туннель должен быть поддержан на стороне клиента (требуется ELTEX\_DATA\_IP(28684)).

Использование отрицательной формы команды (no) удаляет IP-адрес для постройки GRE data туннеля.

**Синтаксис**

```
data-tunnel address <ADDR>
no data-tunnel address
```

**Параметры**

<ADDR> – IP-адрес для построения GRE data туннеля, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

**Значение по умолчанию**

Отсутствует

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-POOL

**Пример**

```
esr(config-pool)# data-tunnel address 192.168.2.66
```

**dead-peer-detection action**

Данной командой устанавливается действие, которое должно предпринять устройство в случае обнаружения недоступности IPsec-соседа механизмом Dead Peer Detection.

**Dead Peer Detection (DPD)** – это механизм проверки состояния и доступности соседних устройств. Механизм периодически отправляет R-U-THERE сообщения (для IKE версии 1) или пустые INFORMATIONAL сообщения (для IKE версии 2) для проверки доступности IPsec-соседа.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
dead-peer-detection action <MODE>
```

```
no dead-peer-detection action
```

**Параметры**

<MODE> – режим работы DPD:

- restart – соединение переустанавливается;
- clear – соединение останавливается;
- hold – соединение поддерживается;
- none – механизм выключен, никаких действий не предпринимается.

**Значение по умолчанию**

none

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IKE-GATEWAY

**Пример**

```
esr(config-ike-gw)# dead-peer-detection action clear
```

**dead-peer-detection interval**

Данной командой устанавливается интервал между отправкой сообщений механизмом DPD. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
dead-peer-detection interval <SEC>
```

```
no dead-peer-detection interval
```

**Параметры**

<SEC> – интервал между отправкой сообщений механизмом DPD, принимает значения [1..180] секунд.

**Значение по умолчанию**

2 секунды

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IKE-GATEWAY

## Пример

```
esr(config-ike-gw)# dead-peer-detection interval 15
```

### dead-peer-detection retransmit jitter

Данной командой устанавливается уровень случайного разброса периода ожидания ответа на сообщения механизма DPD при использовании протокола IKE версии 2.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
dead-peer-detection retransmit jitter <VALUE>
```

```
no dead-peer-detection retransmit jitter
```

### Параметры

<VALUE> – максимальный процент разброса значений, принимает значения [0..100].

### Значение по умолчанию

0 %

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IKE-GATEWAY

## Пример

```
esr(config-ike-gw)# dead-peer-detection retransmit jitter 50
```

### dead-peer-detection retransmit limit

Данной командой устанавливается ограничение максимального периода времени ожидания ответа на сообщения механизма DPD при использовании протокола IKE версии 2.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
dead-peer-detection retransmit limit <SEC>
```

```
no dead-peer-detection retransmit limit
```

### Параметры

<SEC> – максимальный период времени ожидания ответа на сообщения механизма DPD при использовании протокола IKE версии 2, принимает значения [15..300] секунд.

**Значение по умолчанию**

0 секунд

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IKE-GATEWAY

**Пример**

```
esr(config-ike-gw)# dead-peer-detection retransmit limit 180
```

**dead-peer-detection retransmit timeout**

Данной командой устанавливается базовый период времени ожидания ответа на сообщения механизма DPD при использовании протокола IKE версии 2.

Для первого отправленного сообщения DPD период времени ожидания ответа будет равен базовому периоду, указанному в данной команде, а для последующих попыток интервал ожидания будет рассчитан по формуле:

"dead-peer-detection retransmit timeout" \* 1.8 ^ (N-1),

где N – номер попытки, предел которых указан в команде **dead-peer-detection retransmit tries**.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
dead-peer-detection retransmit timeout <SEC>
```

```
no dead-peer-detection retransmit timeout
```

**Параметры**

<SEC> – базовый период времени ожидания ответа на сообщения механизма DPD при использовании протокола IKE версии 2, принимает значения [1..30] секунд.

**Значение по умолчанию**

4 секунды

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IKE-GATEWAY

**Пример**

```
esr(config-ike-gw)# dead-peer-detection retransmit timeout 2
```

### dead-peer-detection retransmit tries

Данной командой устанавливается количество попыток повторной отправки сообщений механизма DPD после наступления таймаута ожидания ответа при использовании протокола IKE версии 2.

Для первого отправленного сообщения DPD период времени ожидания ответа будет равен базовому периоду, указанному в команде **dead-peer-detection retransmit timeout**, а для последующих попыток интервал ожидания будет рассчитан по формуле:

"dead-peer-detection retransmit timeout" \* 1.8 ^ (N-1),

где N – номер попытки, предел которых указан в данной команде.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### Синтаксис

```
dead-peer-detection retransmit tries <TRIES>
```

```
no dead-peer-detection retransmit tries
```

#### Параметры

<TRIES> – количество попыток повторной отправки сообщений механизма DPD в случае наступления таймаута ожидания ответа при использовании протокола IKE версии 2, принимает значения от 1 до 10.

#### Значение по умолчанию

5 попыток

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-IKE-GATEWAY

#### Пример

```
esr(config-ike-gw)# dead-peer-detection retransmit tries 3
```

### dead-peer-detection timeout

Данной командой задаётся период времени ожидания ответа на сообщения механизма DPD при использовании протокола IKE версии 1.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### Синтаксис

```
dead-peer-detection timeout <SEC>
```

```
no dead-peer-detection timeout
```

#### Параметры

<SEC> – период времени для ответа на сообщения механизма DPD, принимает значения [1..180] секунд.

**Значение по умолчанию**

30 секунд

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IKE-GATEWAY

**Пример**

```
esr(config-ike-gw)# dead-peer-detection timeout 60
```

**description**

Команда используется для изменения описания профиля, набора ключей, политики или шлюза протокола IKE.

Использование отрицательной формы команды (no) удаляет описание.

**Синтаксис**

```
description <DESCRIPTION>
```

```
no description
```

**Параметры**

<DESCRIPTION> – описание профиля, задаётся строкой до 255 символов.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IKE-PROPOSAL

CONFIG-IKE-POLICY

CONFIG-IKE-KEYRING

CONFIG-IKE-GATEWAY

**Пример**

```
esr(config-ike-proposal)# description "my proposal"
```

**dh-group**

Данной командой устанавливается номер группы метода Диффи-Хеллмана. Номер группы определяет уровень защищенности IKE-соединения при обмене ключами – защищенность возрастает с ростом номера группы, но увеличивается и время установления соединения.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
dh-group <DH-GROUP>
```

```
no dh-group
```

### Параметры

<DH-GROUP> – номер группы Диффи-Хеллмана, принимает значения [1, 2, 5, 14-31].

### Значение по умолчанию

1

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IKE-PROPOSAL

### Пример

```
esr(config-ike-proposal)# dh-group 5
```

### encryption algorithm

Данной командой выбирается алгоритм шифрования, используемый при установлении IKE-соединения.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
encryption algorithm <ALGORITHM>
```

```
no encryption algorithm
```

### Параметры

<ALGORITHM> – идентификатор протокола шифрования, принимает значения: des, 3des, blowfis28, blowfis92, blowfish256, aes128, aes192, aes256, aes128ctr, aes192ctr, aes256ctr, camellia128, camellia192, camellia256.

### Значение по умолчанию

3des

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IKE-PROPOSAL

## Пример

```
esr(config-ike-proposal)# encryption algorithm aes128
```

## identity

Данной командой устанавливается соответствие идентификатора IPsec клиента и ключа PSK, который будет использован при аутентификации.

Использование отрицательной формы команды (no) удаляет соответствие для указанного идентификатора.

## Синтаксис

```
identity { dns <NAME> | ipv4 <ADDR/LEN> } pre-shared-key { ascii-text { <TEXT> |
encrypted <ENCRYPTED-TEXT>} | hexadecimal { <HEX> | encrypted <ENCRYPTED-HEX> } }
no identity { dns <NAME> | ipv4 <ADDR/LEN> }
```

## Параметры

<NAME> – полное доменное имя хоста (FQDN). Пример записи доменного имени – "router.example.loc". В команде также можно указывать wild-card домены, используя символ "\*", например "\*example.loc";

<ADDR/LEN> – IP-подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32];

<TEXT> – строка [1..64] ASCII-символов;

<HEX> – число размером [1..32] байт, задаётся строкой [2..128] символов в шестнадцатеричном формате (0xYYYY...) или (YYYY...);

<ENCRYPTED\_TEXT> – зашифрованный пароль размером [1..32] байт, задаётся строкой [2..128] символов;

<ENCRYPTED\_HEX> – зашифрованное число размером [2..64] байт, задаётся строкой [2..256] символов.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IKE-KEYRING

## Пример

```
esr(config-ike-keyring)# identity ipv4 98.0.127.37/32 pre-shared-key ascii-text password
esr(config-ike-keyring)# identity dns router.example.loc pre-shared-key ascii-text password
```

## ike-policy

Данной командой устанавливается привязка политики протокола IKE к шлюзу.

Использование отрицательной формы команды (no) удаляет привязку политики.

## Синтаксис

```
[no] ike-policy <NAME>
```



## Параметры

<NAME> – имя политики протокола IKE, задаётся строкой до 31 символа.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IKE-GATEWAY

## Пример

```
esr(config-ike-gw)# ike-policy ike_pol1
```

## ip prefix

Данной командой указывается пул адресов, из которого адреса будут выдаваться IPsec-клиентам.

Использование отрицательной формы команды (no) удаляет пул адресов, из которого адреса будут выдаваться IPsec-клиентам.

## Синтаксис

```
ip prefix <ADDR/LEN>
```

```
no ip prefix
```

## Параметры

<ADDR/LEN> – IP-подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].

## Значение по умолчанию

Не задан.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-POOL

## Пример

```
esr(config-pool)# ip prefix 192.168.0.0/16
```

## keyring

Данной командой указывается набор ключей аутентификации IKE.

Использование отрицательной формы команды (no) удаляет набор ключей.

## Синтаксис

```
keyring <NAME>  
no keyring
```

## Параметры

<NAME> – название набор ключей аутентификации IKE, задаётся строкой до 31 символа.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IKE-POLICY

## Пример

```
esr(config-ike-policy)# keyring ike_keyring
```

## lifetime seconds

Данной командой задаётся время жизни соединения протокола IKE.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
lifetime seconds <SEC>  
no lifetime seconds
```

## Параметры

<SEC> – период времени, принимает значения [4 ..86400] секунд.

## Значение по умолчанию

10800 секунд

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IKE-POLICY

## Пример

```
esr(config-ike-policy)# lifetime 21600
```

## local address

Данной командой устанавливается IP-адрес локального шлюза IPsec-туннеля.

Использование отрицательной формы команды (no) удаляет IP-адрес локального шлюза.

### Синтаксис

```
local address <ADDR>
```

```
no local address
```

### Параметры

<ADDR> – IP-адрес локального шлюза.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IKE-GATEWAY

### Пример

```
esr(config-ike-gw)# local address 192.168.1.1
```

### local id

Данной командой устанавливается идентификатор локального шлюза IPsec-туннеля, использующийся для идентификации в процессе согласования IKE-ключей. Если команда не указана, то при использовании аутентификации по ключам (PSK), в качестве идентификатора локального шлюза используется адрес локального шлюза. При использовании аутентификации по сертификатам X.509, в качестве идентификатора локального шлюза используется значение "Distinguished name" владельца сертификата X.509.

Использование отрицательной формы команды (no) удаляет настройку идентификатора локального шлюза.

### Синтаксис

```
local id { any | dns <NAME> | ipv4 <ADDR> | keyid <KEY> }
```

```
no local id
```

### Параметры

any – ключ, обозначающий "любой" идентификатор;

<NAME> – полное доменное имя хоста (FQDN). Пример записи доменного имени – router.example.loc;

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<KEY> – текстовая строка длиной до 255 символов.

### Необходимый уровень привилегий

10

### Значение по умолчанию

Отсутствует

## Командный режим

### CONFIG-IKE-GATEWAY

#### Пример

```
esr(config-ike-gw)# local id ipv4 192.168.18.1
```

#### local interface

Данной командой устанавливается использование IP-адреса, назначенного на интерфейс в качестве локального шлюза IPsec-туннеля.

При использовании отрицательной формы команды (no) прекращается использование IP-адреса, назначенного на интерфейс в качестве локального шлюза.

#### Синтаксис

```
local interface <IF>
```

```
no local interface
```

#### Параметры

<IF> – тип и идентификатор интерфейса, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

#### Необходимый уровень привилегий

10

## Командный режим

### CONFIG-IKE-GW

#### Пример

```
esr(config-ike-gw)# local interface gigabitethernet 1/0/1
```

#### local network

Данной командой устанавливается IP-адрес подсети отправителя, а также IP-протокол и порт. Трафик, удовлетворяющий заданным критериям, будет направлен в IPsec-туннель.

Использование отрицательной формы команды (no) удаляет IP-адрес подсети отправителя.

#### Синтаксис

```
[no] local network { <ADDR/LEN> | dynamic } [ protocol { <TYPE> | <ID> } [ port <PORT> ] ]
```

#### Параметры

<ADDR/LEN> – IP-подсеть отправителя. Параметр задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32];

dynamic – в качестве IP-адрес подсети отправителя будет использован IP-адрес, с которого устанавливается IPsec-соединение;

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;

<ID> – идентификационный номер IP-протокола, принимает значения [0x00-0xFF];

<PORT> – TCP/UDP-порт, принимает значения [1..65535].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IKE-GATEWAY

### Пример

```
esr(config-ike-gw)# local network 192.168.1.0/24 protocol tcp port 22
```

### management-tunnel address

Данной командой устанавливается IP-адрес туннеля для постройки GRE management туннеля, передаваемого клиенту, подключаемому через IPsec с использованием динамического конфигурирования параметров. GRE management туннель должен быть поддержан на стороне клиента (требуется ELTEX\_MANAGEMENT\_IP(28683)).

Использование отрицательной формы команды (no) удаляет IP-адрес туннеля для постройки GRE management туннеля.

### Синтаксис

```
management-tunnel address <ADDR>
no management-tunnel address
```

### Параметры

<ADDR> – IP-адрес для постройки GRE management туннеля, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-POOL

### Пример

```
esr(config-pool)# management-tunnel address 192.168.2.87
```

**mobike disable**

Данная команда отключает расширение MOBIKE IKEv2, которое позволяет инициатору ike-сессии изменять local address в соответствии с RFC 4555.

Использование отрицательной формы команды (no) активирует функцию автоматического выбора local address при недоступности описанного в конфигурации.

**Синтаксис**

```
[ no ] mobike disable
```

**Параметры**

Отсутствуют.

**Значение по умолчанию**

Включено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IKE-GATEWAY

**Пример**

```
esr(config-ike-gateway)# mobike disable
```

**mode**

Данной командой устанавливается режим согласования первой фазы протокола IKE.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
mode <MODE>
```

```
no mode
```

**Параметры**

<MODE> – режим первой фазы IKE, принимает значения:

- main – состоит из трех двусторонних обменов между отправителем и получателем:
  - Согласуются алгоритмы аутентификации и шифрования, которые будут использоваться для защиты IKE-соединения посредством сопоставления профилей протокола IKE каждого узла;
  - Используя алгоритм Диффи-Хеллмана, стороны обмениваются общим секретным ключом. Также узлы проверяют идентификацию друг друга путем передачи и подтверждения последовательности псевдослучайных чисел;
  - Проверяется идентичность противоположной стороны. В результате выполнения основного режима создается безопасный канал для второй фазы протокола IKE.
- aggressive – этот режим обходится меньшим числом обменов и, соответственно, числом пакетов:

- В первом сообщении (от инициатора) отправляется информация, которая используется для установления IKE-соединения: предложение параметров SA, инициирование обмена Диффи-Хеллмана, отправление псевдослучайного числа и идентификатора пакета;
- Во втором сообщении ответчик принимает SA, аутентифицирует инициатора, отправляет псевдослучайное число и свой IKE-идентификатор;
- В третьем сообщении инициатор аутентифицирует ответчика и подтверждает обмен.

### Значение по умолчанию

main

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IKE-POLICY

### Пример

```
esr(config-ike-policy)# mode aggressive
```

### mode

Данной командой устанавливается режим перенаправления трафика в туннель.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
mode <MODE>
```

```
no mode
```

### Параметры

<MODE> – режим перенаправления трафика в туннель, принимает значения:

- policy-based – трафик перенаправляется на основе принадлежности к указанным в политиках подсетям;
- route-based – трафик перенаправляется на основе маршрутов, шлюзом у которых является туннельный интерфейс.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IKE-GATEWAY

### Пример

```
esr(config-ike-gw)# mode route-based
```

**password**

Данная команда используется для установки пароля пользователя для IKE-GETWAY. Пароль может быть задан как в открытом виде, так и в виде хеш sha512.

Использование отрицательной формы команды (no) удаляет пароль пользователя для IKE-GETWAY из системы.

**Синтаксис**

```
password ascii-text { <CLEAR-TEXT> | encrypted <HASH_SHA512> }
no password
```

**Параметры**

<CLEAR-TEXT> – пароль, задаётся строкой [8..32] символов, принимает значения [0-9a-fA-F].

<HASH\_SHA512> – хеш пароля по алгоритму sha512, задаётся строкой до 110 символов.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-PROFILE

**Пример**

```
esr(config-profile) password tteesstt
```

**password local-crt-key**

Данная команда используется для установки пароля от зашифрованной цепочки сертификатов (сертификаты назначаются при помощи команды [crypto](#)).

Использование отрицательной формы команды (no) удаляет пароль.

**Синтаксис**

```
password local-crt-key ascii-text { <CLEAR-TEXT> | encrypted <HASH_SHA512> }
no password local-crt-key
```

**Параметры**

<CLEAR-TEXT> – пароль, задаётся строкой [8..32] символов, принимает значения [0-9a-fA-F].

<HASH\_SHA512> – хеш пароля по алгоритму sha512, задаётся строкой до 110 символов.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IKE-POLICY



## Пример

```
esr(config-ike-policy) password tteesstt
```

## pfs dh-group

Данной командой устанавливается номер группы метода Диффи-Хеллмана. Номер группы определяет уровень защищенности IPsec-соединения при обмене ключами – защищенность возрастает с ростом номера группы, но увеличивается и время установления соединения.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
pfs dh-group <DH-GROUP>
```

```
no pfs dh-group
```

## Параметры

<DH-GROUP> – номер группы Диффи-Хеллмана, принимает значения [1, 2, 5, 14-31].

## Значение по умолчанию

1

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IPSEC-PROPOSAL

## Пример

```
esr(config-isek-proposal)# pfs dh-group 5
```

## pre-shared-key

Данной командой устанавливается общий секретный ключ для аутентификации, должен совпадать у обеих сторон, устанавливающих туннель.

Использование отрицательной формы команды (no) удаляет установленный ключ.

## Синтаксис

```
pre-shared-key { ascii-text { <TEXT> | encrypted <ENCRYPTED-TEXT> } | hexadecimal { <HEX> | encrypted <ENCRYPTED-HEX> } }
```

```
no pre-shared-key
```

## Параметры

<TEXT> – строка [1..64] ASCII-символов;

<HEX> – число размером [1..32] байт, задаётся строкой [2..128] символов в шестнадцатеричном формате (0xYYYY...) или (YYYY...);

<ENCRYPTED-TEXT> – зашифрованный пароль размером [1..32] байт, задаётся строкой [2..128] символов;

<ENCRYPTED-HEX> – зашифрованное число размером [2..64] байт, задаётся строкой [2..256] символов.

### Значение по умолчанию

none

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IKE-POLICY

### Пример

```
esr(config-ike-policy)# pre-shared-key hexadecimal abc123
```

### proposal

Данной командой устанавливается привязка профиля протокола IKE к политике.

Использование отрицательной формы команды (no) удаляет привязку профиля протокола IKE.

### Синтаксис

```
[no] proposal <NAME>
```

### Параметры

<NAME> – имя профиля протокола IKE, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IKE-POLICY

### Пример

```
esr(config-ike-policy)# proposal ike_prop1
```

### remote address

Данной командой устанавливается IP-адрес удаленного шлюза IPsec-туннеля.

Использование отрицательной формы команды (no) удаляет IP-адрес удаленного шлюза.

**Синтаксис**

```
remote address { <ADDR> | any }
no remote address
```

**Параметры**

<ADDR> – IP-адрес удаленного шлюза.

any – ключ, позволяющий принимать запросы на установление IKE-сессии от любого IP-адреса.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IKE-GATEWAY

**Пример**

```
esr(config-ike-gw)# remote address 192.168.1.2
```

**remote id**

Данной командой устанавливается ожидаемый идентификатор удаленного шлюза IPsec-туннеля, получаемый от удаленной стороны для идентификации в процессе согласования IKE-ключей.

Использование отрицательной формы команды (no) удаляет настройку ожидаемого идентификатора удаленного шлюза.

**Синтаксис**

```
remote id { any | dns <NAME> | ipv4 <ADDR> | keyid <KEY> }
no remote id
```

**Параметры**

any – ключ, обозначающий "любой" идентификатор;

<NAME> – полное доменное имя хоста (FQDN). Пример записи доменного имени – router.example.loc;

<ADDR> – IP-адрес, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<KEY> – текстовая строка длиной до 255 символов.

**Необходимый уровень привилегий**

10

**Значение по умолчанию**

Отсутствует

**Командный режим**

CONFIG-IKE-GATEWAY

## Пример

```
esr(config-ike-gw)# remote id dns router.example.loc
```

## remote network

Данной командой устанавливается IP-адрес подсети получателя, а также IP-протокол и порт или назначается динамический пул адресов для удалённых клиентов, использующих XAUTH. Трафик, удовлетворяющий заданным критериям, будет направлен в IPsec-туннель.

Использование отрицательной формы команды (no) удаляет IP-адрес подсети отправителя.

## Синтаксис

```
remote network { dynamic pool <POOL> | <ADDR/LEN> [ protocol { <TYPE> | <ID> } [ port <PORT> ] ] | any }
```

```
no remote network { dynamic pool | <ADDR/LEN> [ protocol { <TYPE> | <ID> } [ port <PORT> ] ] | any }
```

## Параметры

<POOL> – выделенный динамический пул адресов для клиентов XAUTH;

<ADDR/LEN> – IP-подсеть получателя. Параметр задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32];

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;

<ID> – идентификационный номер IP-протокола, принимает значения [0x00-0xFF];

<PORT> – TCP/UDP-порт, принимает значения [1..65535];

any – ключ, указывающий на необходимость шифрования любого исходящего трафика.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IKE-GATEWAY

## Пример

```
esr(config-ike-gw)# remote network 192.168.0.0/24 protocol tcp port 22
```

## remote network dynamic client

Данной командой включается получение списка удаленных сетей от IPsec-VPN-сервера.

При использовании отрицательной формы команды (no) отключается получение списка удаленных сетей от IPsec-VPN-сервера.

## Синтаксис

```
[no] remote network dynamic client
```

**Параметры**

Отсутствуют.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IKE-GW

**Пример**

```
esr(config-ike-gw)# remote network dynamic client
```

**security ike gateway**

Данной командой осуществляется переход в командный режим конфигурирования шлюза IKE SECURITY IKE GATEWAY. Если шлюз IKE с указанным именем не существует в конфигурации, то он будет создан. Параметры шлюза включают в себя VTI-интерфейс, в который будет направляться трафик, политика и версия протокола IKE, а также режим перенаправления трафика в туннель.

Использование отрицательной формы команды (no) удаляет шлюз протокола IKE.

**Синтаксис**

```
[no] security ike gateway <NAME>
```

**Параметры**

<NAME> – имя шлюза протокола IKE, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все IKE-шлюзы.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# security ike gateway ike_gw1
esr(config-ike-gw)#
```

**security ike keyring**

Данной командой создается набор ключей аутентификации для протокола IKE, который включает в себя наборы соответствий адресов удаленных IPsec-клиентов и ключей PSK, которые будут использоваться при аутентификации.

Использование отрицательной формы команды (no) удаляет шлюз протокола IKE.

**Синтаксис**

```
[no] security ike keyring <NAME>
```

**Параметры**

<NAME> – имя шлюза протокола IKE, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все наборы.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# security ike keyring ike_keyring1
esr(config-ike-gw)#
```

**security ike policy**

Данной командой создается политика IKE, которая включает в себя профили протокола IKE, общий секретный ключ для аутентификации и режим согласования первой фазы протокола IKE.

Использование отрицательной формы команды (no) удаляет указанную политику. Команда устанавливает режим командной строки SECURITY IKE POLICY.

**Синтаксис**

```
[no] security ike policy <NAME>
```

**Параметры**

<NAME> – имя политики IKE, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все IKE-политики.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

## Пример

```
esr(config)# security ike policy ike_pol1
esr(config-ike-policy)#
```

## security ike proposal

Данной командой создается профиль протокола IKE (Internet Key Exchange), который включает в себя параметры алгоритмов шифрования и аутентификации метода Диффи-Хеллмана, которые будут использоваться при согласовании параметров IKE со встречной стороной VPN соединения при создании Security Association (SA). Кроме того, профиль задаёт предельное время действия SA. Использование отрицательной формы команды (no) удаляет заданный профиль.

## Синтаксис

```
[no] security ike proposal <NAME>
```

## Параметры

<NAME> – имя профиля протокола IKE, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все IKE-профили.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# security ike proposal ike_prop1
esr(config-ike-proposal)#
```

## security ike session reauthentication

Данной командой задается режим повторной аутентификации ассоциации безопасности IKE в протоколе IKE версии 2.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
security ike session reauthentication <MODE>
no security ike session reauthentication
```

## Параметры

<MODE> – режим повторной аутентификации ассоциации безопасности IKE в протоколе IKE версии 2, принимает следующие значения:

- break-before-make – в процессе повторной аутентификации текущая ассоциация безопасности IKE совместно с ассоциациями безопасности IPsec будет разрушена до того, как будет установлена новая ассоциация безопасности IKE.

- **make-before-break** – в процессе повторной аутентификации текущая ассоциация безопасности IKE совместно с ассоциациями безопасности IPsec будет существовать и использоваться до полного завершения процесса установления новой ассоциации безопасности IKE.

### Значение по умолчанию

break-before-make

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# security ike session reauthentication make-before-break
```

### security ike session uniqueids

Данной командой задается режим переподключения клиентов XAUTH с одним логином/паролем.

Использование отрицательной формы команды (**no**) устанавливает значение по умолчанию.

### Синтаксис

```
security ike session uniqueids <MODE>
```

```
no security ike session uniqueids
```

### Параметры

<MODE> – режим переподключения, принимает следующие значения:

- **no** – установленное подключение XAUTH будет удалено, если для нового подключения XAUTH инициатором соединения будет отправлено уведомление "INITIAL\_CONTACT", будет назначен ранее использованный IP-адрес. В противном случае, установленное соединение XAUTH будет удержано. Для нового подключения XAUTH будет назначен новый IP-адрес.
- **never** – установленное подключение XAUTH будет удержано. Для нового подключения XAUTH будет назначен новый IP-адрес. Уведомление "INITIAL\_CONTACT" будет в любом случае проигнорировано.
- **replace** – установленное подключение XAUTH будет удалено. Для нового подключения XAUTH будет использован ранее использованный IP-адрес.
- **keep** – установленное подключение XAUTH будет удержано. Новое подключение XAUTH будет отклонено.

### Значение по умолчанию

never

### Необходимый уровень привилегий

10

### Командный режим

CONFIG



## Пример

```
esr(config)# security ike session uniqueids replace
```

## show security ike

Команда используется для просмотра списка шлюзов, политик или профилей.

## Синтаксис

```
show security ike { gateway | policy | proposal } [<NAME>]
```

## Параметры

gateway – при указании команды «gateway» будет выведен список сконфигурированных шлюзов;

policy – при указании команды «policy» будет выведен список сконфигурированных политик;

proposal – при указании команды «proposal» будет выведен список сконфигурированных профилей;

<NAME> – имя. При указании определенного имени шлюза, политики, профиля будет выведена подробная информация.

## Необходимый уровень привилегий

10

## Командный режим

ROOT

## Пример

```
esr# show security ike proposal
Proposal
~~~~~
Name          Auth          Encryption          DH   Hash          Lifetime
-----
aaa           pre-sha      3des                1   sha1          3600
              red-key

esr# show security ike policy
Policy
~~~~~
Name          Mode          Proposal
-----
ike_pol1     main          ike_prop1

esr# show security ike gateway ik_gw
Description:      --
IKE Policy:       ike_pol1
IKE Version:     v1-only
Mode:            route-based
Binding interface: vti1
IKE Dead Peer Detection:
Action:          none
Interval:        2
Timeout:         30
```

**user**

Данной командой задается имя пользователя для аутентификации IKE-GETWAY.

Использование отрицательной формы команды (no) удаляет указанного пользователя.

После выполнения данной команды маршрутизатор переходит в режим конфигурирования пароля пользователя (config-profile).

**Синтаксис**

```
[no] user <NAME>
```

**Параметры**

<NAME> – имя пользователя, задаётся строкой до 31 символа.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-ACCESS-PROFILE

**Пример**

```
esr(config-access-profile)# user connecter963
```

**version**

Данной командой задаётся версия протокола IKE.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
version <VERSION>
```

```
no version
```

**Параметры**

<version> – версия IKE-протокола: v1-only или v2-only.

**Значение по умолчанию**

v1-only

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IKE-GATEWAY

**Пример**

```
esr(config-ike-gw)# version v2-only
```

**Управление VPN. Настройки IPsec****authentication algorithm**

Данной командой устанавливается алгоритм аутентификации. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
authentication algorithm <ALGORITHM>
```

```
no authentication algorithm
```

**Параметры**

<ALGORITHM> – алгоритм аутентификации, принимает значения: md5, sha1, sha2-256, sha2-384, sha2-512.

**Значение по умолчанию**

sha1

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IPSEC-PROPOSAL

**Пример**

```
esr(config-ipsec-proposal)# authentication algorithm md5
```

**clear security ipsec vpn**

Данной командой осуществляется сброс одного из текущих VPN-соединений.

**Синтаксис**

```
clear security ipsec vpn [ vrf <VRF> ] <NAME>
```

**Параметры**

<NAME> – имя VPN, задаётся строкой до 31 символа;

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа.

**Необходимый уровень привилегий**

15

## Командный режим

ROOT

### Пример

```
esr# clear security ipsec vpn IPSEC_MAIN_OFFICE
```

### description

Данной командой выполняется изменение описания.

Использование отрицательной формы команды (no) удаляет описание.

### Синтаксис

```
description <DESCRIPTION>
```

```
no description
```

### Параметры

<DESCRIPTION> – описание профиля, задаётся строкой до 255 символов.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-IPSEC-VPN

CONFIG-IPSEC-PROPOSAL

CONFIG-IPSEC-POLICY

### Пример

```
esr(config-ipsec-vpn)# description "VPN to Moscow Office"
```

### enable

Данной командой активируется IPsec VPN.

Использование отрицательной формы команды (no) деактивирует IPsec VPN.

### Синтаксис

```
[no] enable
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Выключено

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IPSEC-VPN

**Пример**

```
esr(config-ipsec-vpn)# enable
```

**encryption algorithm**

Данной командой устанавливается алгоритм шифрования. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
encryption algorithm <ALGORITHM>
```

```
no encryption algorithm
```

**Параметры**

<ALGORITHM> – протокол шифрования, принимает значения: null, des, 3des, blowfis28, blowfish192, blowfish256, aes128, aes192, aes256, aes128ctr, aes192ctr, aes256ctr, camellia128, camellia192, camellia256.

**Значение по умолчанию**

3des

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IPSEC-PROPOSAL

**Пример**

```
esr(config-ipsec-proposal)# encryption algorithm blowfish128
```

**ike dscp**

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов IKE-протокола.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

**Синтаксис**

```
ike dscp <DSCP>
```

```
no ike dscp
```

### Параметры

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

### Значение по умолчанию

63

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IPSEC-VPN

### Пример

```
esr(config-ipsec-vpn)# ike dscp 40
```

### ike establish-tunnel

Командой устанавливается режим активации VPN. Данная команда актуальна, только если в VPN выбран режим согласования ключей «ike». Настройка режима согласования ключей описана в [mode](#).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ike establish-tunnel <MODE>
```

```
no Ike establish-tunnel
```

### Параметры

<MODE> – режим активации VPN:

- by-request – соединение активируется встречной стороной;
- route – соединение активируется при появлении трафика, маршрутизируемого в туннель;
- immediate – туннель активируется автоматически после применения конфигурации. Если произойдет закрытие соединения от удаленной стороны VPN при использовании режима immediate, тогда установить соединение повторно будет возможно только после перезагрузки маршрутизатора или перезапуске VPN через CLI маршрутизатора.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IPSEC-VPN

### Пример

```
esr(config-ipsec-vpn)# ike establish-tunnel route
```

## ike gateway

Данной командой осуществляется привязка IKE-шлюза к VPN. Данная команда актуальна, только если в VPN выбран режим согласования ключей «ike». Настройка режима согласования ключей описана в [mode](#).

### Синтаксис

```
ike gateway <NAME>
```

```
no ike gateway
```

### Параметры

<NAME> – имя IKE-шлюза, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IPSEC-VPN

### Пример

```
esr(config-ipsec-vpn)# ike gateway ike_gw1
```

## ike idle-time

Данной командой устанавливается значение временного интервала в секундах, по истечению которого соединение закрывается, если не было принято или передано ни одного пакета через SA.

Использование отрицательной формы команды (no) отключает данный таймер.

### Синтаксис

```
ike idle-time <TIME>
```

```
no ike idle-time
```

### Параметры

<TIME> – интервал в секундах, принимает значения [4..86400].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IPSEC-VPN

### Пример

```
esr(config-ipsec-vpn)# ike idle-time 3600
```

**ike rekey disable**

Данной командой отключается пересогласование ключей до разрыва IKE-соединения по истечению времени, количеству переданных пакетов или байт.

Использование отрицательной формы команды (no) включает пересогласование ключей.

**Синтаксис**

```
[no] ike rekey disable
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IPSEC-VPN

**Пример**

```
esr(config-ipsec-vpn)# ike rekey disable
```

**ike rekey margin**

Данной командой можно настроить начало пересогласования ключей IKE-соединения до истечения времени жизни.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
Ike rekey margin { seconds <SEC> | packets <PACKETS> | kilobytes <KB> }
no ike rekey margin { seconds | packets | kilobytes }
```

**Параметры**

<SEC> – интервал времени в секундах, оставшийся до закрытия соединения (задается командой `lifetime seconds`, см. [lifetime](#)). Принимает значения [4..86400].

<PACKETS> – количество пакетов, оставшихся до закрытия соединения (задается командой `lifetime packets`, см. [lifetime](#)). Принимает значения [4..86400].

<KB> – объем трафика в килобайтах, оставшийся до закрытия соединения (задается командой `lifetime kilobytes`, см. [lifetime](#)). Принимает значения [4..86400].

**Значение по умолчанию**

Пересогласование ключей до истечения времени – за 540 секунд.



Пересогласование ключей до истечения объема трафика и количества пакетов – отключено.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IPSEC-VPN

### Пример

```
esr(config-ipsec-vpn)# ike rekey margin seconds 1800
```

### ike rekey randomization

Данной командой устанавливается уровень случайного разброса значений параметров `margin seconds`, `margin packets`, `margin kilobytes`.

Использование отрицательной формы команды (`no`) устанавливает значение по умолчанию.

### Синтаксис

```
ike rekey randomization <VALUE>
```

```
no ike rekey randomization
```

### Параметры

<VALUE> – максимальный процент разброса значений, принимает значения [1..100].

### Значение по умолчанию

100%

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IPSEC-VPN

### Пример

```
esr(config-ipsec-vpn)# ike rekey randomization 10
```

### ike ipsec-policy

Данная команда устанавливает привязку IPsec-политики к VPN. Данная команда актуальна, только если в VPN выбран режим согласования ключей «ike». Настройка режима согласования ключей описана в [mode](#).

### Синтаксис

```
ike ipsec-policy <NAME>
```

```
no ike ipsec-policy
```

### Параметры

<NAME> – имя IPsec-политики, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IPSEC-VPN

### Пример

```
esr(config-ipsec-vpn)# ike ipsec-policy ipsec_pol1
```

### lifetime

Данной командой устанавливается время жизни IPsec-туннеля.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
lifetime { seconds <SEC> | packets <PACKETS> | kilobytes <KB> }
no lifetime { seconds | packets | kilobytes }
```

### Параметры

<SEC> – период времени жизни IPsec-туннеля, по истечении происходит пересогласование. Принимает значения [1140..86400] секунд;

<PACKETS> – количество пакетов, после передачи которого происходит пересогласование IPsec-туннеля. Принимает значения [4..86400];

<KB> – объем трафика, после передачи которого происходит пересогласование IPsec-туннеля. Принимает значения [4..4608000] секунд.

### Значение по умолчанию

3600 секунд

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IPSEC-POLICY

### Пример

```
esr(config-ipsec-proposal)# lifetime seconds 3600
```

## manual authentication algorithm

Данной командой устанавливается алгоритм аутентификации. Данная команда актуальна, только если в VPN выбран режим согласования ключей «manual». Настройка режима согласования ключей описана в [mode](#).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
manual authentication algorithm <ALGORITHM>
no manual authentication algorithm
```

### Параметры

<ALGORITHM> – алгоритм аутентификации, принимает значения [md5, md5-128, sha1, sha1-160, aesxcbc, sha2-256, sha2-384, sha2-512].

### Значение по умолчанию

none

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IPSEC-VPN

### Пример

```
esr(config-ipsec-vpn)# manual authentication algorithm sha1
```

## manual authentication key

Данной командой устанавливается ключ аутентификации. Данная команда актуальна, только если в VPN выбран режим согласования ключей «manual». Настройка режима согласования ключей описана в [mode](#).

### Синтаксис

```
manual authentication key { ascii-text {<TEXT> | encrypted <ENCRYPTED-TEXT>} |
hexadecimal {<HEX> | encrypted <ENCRYPTED-HEX> } }
no manual authentication key
```

### Параметры

<TEXT> – строка [1..64] ASCII-символов;

<HEX> – число размером [1..32] байт, задаётся строкой [2..128] символов в шестнадцатеричном формате (0xYYYY...) или (YYYY...);

<ENCRYPTED\_TEXT> – зашифрованный пароль размером [1..32] байт, задаётся строкой [2..128] символов;

<ENCRYPTED\_HEX> – зашифрованное число размером [2..64] байт, задаётся строкой [2..256] символов.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IPSEC-VPN

## Пример

```
esr(config-ipsec-vpn)# manual authentication key hexadecimal abcdef
```

## manual bind-interface vti

Данной командой указывается туннельный интерфейс, через который будет проходить трафик в режиме туннеля route-based. Данная команда актуальна, только если в VPN выбран режим согласования ключей «manual». Настройка режима согласования ключей описана в [mode](#).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
manual bind-interface vti <VTI>
```

```
no manual bind-interface vti
```

## Параметры

<VTI> – индекс интерфейса VTI, принимает значения:

- ESR-10/12V/12VF/15/15R/15VF – [1..10];
- ESR-20/21/30/31/100/200 – [1..250];
- ESR-1000/1200/1500/1511/1700/3100/3200/3200L/3300 – [1..500].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IPSEC-VPN

## Пример

```
esr(config-ipsec-vpn)# manual bind-interface vti 0
```

## manual encryption algorithm

Данной командой устанавливается алгоритм шифрования. Данная команда актуальна, только если в VPN выбран режим согласования ключей «manual». Настройка режима согласования ключей описана в [mode](#).

Использование отрицательной формы команды (no) удаляет установленное значение.

**Синтаксис**

```
manual encryption algorithm <ALGORITHM>
```

```
no manual encryption algorithm
```

**Параметры**

<ALGORITHM> – алгоритм шифрования, принимает значения: des, 3des, blowfis28, blowfis92, blowfish256, aes128, aes192, aes256, aes128ctr, aes192ctr, aes256ctr, camellia128, camellia192, camellia256.

**Значение по умолчанию**

3des

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IPSEC-VPN

**Пример**

```
esr(config-ipsec-vpn)# manual encryption algorithm blowfis28
```

**manual encryption key**

Данной командой устанавливается ключ шифрования. Данная команда актуальна, только если в VPN выбран режим согласования ключей «manual». Настройка режима согласования ключей описана в [mode](#).

Использование отрицательной формы команды (no) удаляет установленное значение.

**Синтаксис**

```
manual encryption key { ascii-text { < TEXT> | encrypted <ENCRYPTED-TEXT> } | hexadecimal  
{ <HEX> | encrypted <ENCRYPTED-HEX> } }
```

```
no manual encryption key
```

**Параметры**

<TEXT> – строка [1..36] ASCII-символов;

<HEX> – число размером [1..24] байт, задаётся строкой [2..72] символов в шестнадцатеричном формате (0xYYYY...) или (YYYY...);

<ENCRYPTED-TEXT> – зашифрованный пароль размером [1..24] байт, задаётся строкой [2..72] символов;

<ENCRYPTED-HEX> – зашифрованное число размером [2..36] байт, задаётся строкой [2..144] символов.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IPSEC-VPN

## Пример

```
esr(config-ipsec-vpn)# manual encryption key hexadecimal 0x123456
```

### manual mode

С помощью данной команды осуществляется установка режима перенаправления трафика в туннель. Данная команда актуальна, только если в VPN выбран режим согласования ключей «manual». Настройка режима согласования ключей описана в [mode](#).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
manual mode <MODE>
```

```
no manual mode
```

### Параметры

<MODE> – режим прохождения трафика:

- policy-based – трафик перенаправляется на основе принадлежности к указанным в политиках подсетям;
- route-based – трафик перенаправляется на основе маршрутов, у которых шлюзом является туннельный интерфейс.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IPSEC-VPN

## Пример

```
esr(config-ipsec-vpn)# manual mode route-based
```

### manual protocol

Данной командой устанавливается инкапсулирующий протокол. Данная команда актуальна, только если в VPN выбран режим согласования ключей «manual». Настройка режима согласования ключей описана в [mode](#).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
manual protocol <TYPE>
```

```
no manual protocol
```

### Параметры

<TYPE> – тип протокола, принимает значения:

- ah – данный протокол осуществляет только аутентификацию трафика, шифрование данных не выполняется;
- esp – данный протокол осуществляет аутентификацию и шифрование трафика.

### Значение по умолчанию

esp

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IPSEC-VPN

### Пример

```
esr(config-ipsec-vpn)# manual protocol ah
```

### manual spi

Данной командой устанавливается индекс параметров безопасности. Данная команда актуальна, только если в VPN выбран режим согласования ключей «manual». Настройка режима согласования ключей описана в [mode](#).

Использование отрицательной формы команды (no) удаляет индекс параметров безопасности.

### Синтаксис

```
manual spi <HEX>
```

```
no manual spi
```

### Параметры

<HEX> – индекс параметров безопасности, задаётся значение размером 32 бита (8 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IPSEC-VPN

### Пример

```
esr(config-ipsec-vpn)# manual spi FF
```

### mode

Данной командой устанавливается режим согласования данных, необходимых для активации VPN.

**Синтаксис**

```
mode <MODE>
```

```
no mode
```

**Параметры**

<MODE> – режим работы VPN:

- `ike` – согласование алгоритмов аутентификации и шифрования, ключей аутентификации и шифрования, индекса параметра безопасности и других данных осуществляется через протокол IKE;
- `manual` – пользователь должен сам настроить идентичные параметры на обоих узлах для работы VPN. При данном режиме не происходит установления IKE-соединения между узлами. Каждый из узлов шифрует и дешифрует пакеты, основываясь только на заданных параметрах.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IPSEC-VPN

**Пример**

```
esr(config-ipsec-vpn)# mode ike
```

**proposal**

Данной командой к политике привязываются профили набора протоколов IPsec.

Использование отрицательной формы команды (`no`) удаляет привязку к указанному профилю.

**Синтаксис**

```
[no] proposal <NAME>
```

**Параметры**

<NAME> – имя профиля набора протоколов IPsec, задаётся строкой до 31 символа.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IPSEC-POLICY

**Пример**

```
esr(config-ipsec-policy)# proposal ipsec_prop1
```



**protocol**

Данной командой устанавливается инкапсулирующий протокол.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
protocol <PROTOCOL>
```

```
no protocol
```

**Параметры**

<PROTOCOL> – инкапсулирующий протокол, принимает значения:

- ah – данный протокол осуществляет только аутентификацию трафика, шифрование данных не выполняется;
- esp – данный протокол осуществляет аутентификацию и шифрование трафика.

**Значение по умолчанию**

```
esp
```

**Необходимый уровень привилегий**

```
15
```

**Командный режим**

```
CONFIG-IPSEC-PROPOSAL
```

**Пример**

```
esr(config-ipsec-proposal)# protocol ah
```

**security ipsec policy**

Данной командой создается политика набора протоколов IPsec, которая включает в себя профили набора протоколов IPsec для согласования второй фазы протокола IKE.

Использование отрицательной формы команды (no) удаляет установленное значение.

Команда устанавливает режим командной строки SECURITY IPSEC POLICY.

**Синтаксис**

```
[no] security ipsec policy <NAME>
```

**Параметры**

<NAME> – имя политики IPsec, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все IPsec-политики.

**Необходимый уровень привилегий**

```
10
```

**Командный режим**

CONFIG

**Пример**

```
esr(config)# security ipsec policy ipsec_pol1
esr(config-ipsec-policy)#
```

**security ipsec proposal**

Данной командой создается профиль для набора протоколов IPsec. Профиль IPsec включает в себя параметры алгоритмов шифрования и аутентификации, протокола защиты соединения IPsec-туннеля, а также время жизни соединения.

Использование отрицательной формы команды (no) удаляет указанный профиль.

Команда устанавливает режим командной строки SECURITY IPSEC PROPOSAL.

**Синтаксис**

```
[no] security ipsec proposal <NAME>
```

**Параметры**

<NAME> – имя профиля IPsec, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все IPsec-профили.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# security ipsec proposal ipsec_prop1
esr(config-ipsec-proposal)#
```

**security ipsec vpn**

Данной командой создается VPN на основе набора протоколов IPsec и устанавливается командный режим SECURITY IPSEC VPN.

Использование отрицательной формы команды (no) удаляет сконфигурированный VPN.

**Синтаксис**

```
[no] security ipsec vpn <NAME>
```

**Параметры**

<NAME> – имя VPN, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все VPN.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# security ipsec vpn ipsec_vpn1
esr(config-ipsec-vpn)#
```

**show security ipsec**

Данной командой выполняется просмотр конфигураций VPN, политик и профилей набора протоколов IPsec.

**Синтаксис**

```
show security ipsec { vpn configuration | policy | proposal } [<NAME>]
```

**Параметры**

vpn configuration – при указании данной команды будет выведена конфигурация всех VPN;

policy – при указании данной команды будет выведен список сконфигурированных политик набора протоколов IPsec;

proposal – при указании команды будет выведен список сконфигурированных профилей набора протоколов IPsec;

<NAME> – имя. При указании определенного имени VPN, политики или профиля будет выведена подробная информация.

**Необходимый уровень привилегий**

10

**Командный режим**

ROOT

**Пример**

```
esr# show security ipsec proposal
  Proposal
  ~~~~~~
Name          Prot   Enc. alg.      Auth. alg.      Lifetime
-----
ipsec_prop1   esp   aes128         sha1             28800 sec
esr# show security ipsec policy
Name          Description      Proposal
-----
ipsec_pol1                    ipsec_prop1
Master# show security ipsec vpn configuration IPSECVPN
Description:      --
```

```

State:                               Enabled
IKE:
  Establish tunnel:                   immediate
  IPsec policy:                       IPSECPOLICY
  IKE gateway:                        IKEGW
  IKE DSCP:                            63
  IKE idle-time:                       0s
  IKE rekeying:                        Enabled
  Margin time:                         540s
  Margin kilobytes:                    0
  Margin packets:                      0
  Randomization:                       100%

```

### show security ipsec vpn authentication

Данная команда позволяет посмотреть список и параметры подключившихся IPsec-VPN-клиентов.

#### Синтаксис

```
show security ipsec vpn authentication <NAME> [ vrf <VRF> ]
```

#### Параметры

<NAME> – имя созданного IPsec VPN, задаётся строкой до 31 символа.

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет включено разрешение DNS-имен.

#### Необходимый уровень привилегий

10

#### Командный режим

ROOT

#### Пример

```

esr# show security ipsec vpn authentication
Local host      Remote host      Local subnet      Remote subnet      Authentication
State
-----
2.2.2.1         2.2.2.2         192.168.2.0/24   192.168.1.1/32    Xauth PSK,
login: ipsec                    Established

```

### show security ipsec vpn status

Данной командой выполняется просмотр статуса всех VPN, которые устанавливают соединение через IKE-протокол либо определенного VPN при указании его имени.

#### Синтаксис

```
show security ipsec vpn status [ vrf <VRF> ] [ <NAME> ]
```

**Параметры**

<NAME> – имя VPN, задаётся строкой до 31 символа;

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа.

**Необходимый уровень привилегий**

10

**Командный режим**

ROOT

**Пример**

```
esr# show security ipsec vpn status
Name          Local host  Remote host  Initiator spi  Responder spi  State
-----
ipsec_vpn1    10.100.14.1 10.100.14.2 0x05d8e0ac3543f0cb 0xcfa1c4179d001154 Established
```

**type**

Данной командой задается тип инкапсуляции IPsec.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
type <TYPE>
no type
```

**Параметры**

<TYPE> – тип инкапсуляции IPsec, принимает значения:

- tunnel;
- transport.

**Значение по умолчанию**

tunnel

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IPSEC-VPN

**Пример**

```
esr(config-access-vpn)# type transport
```

## Управление VPN. Настройки удаленного доступа

- Общие команды настройки удаленного доступа
  - authentication required disable
  - clear remote-access counters
  - clear remote-access session
  - description
  - enable
  - encryption mppe
  - remote-access
  - show remote-access configuration
  - show remote-access counters
  - show remote-access status
- Настройка L2TP over IPsec/PPTP-сервера
  - authentication method
  - authentication mode
  - dns-servers
  - dscp
  - ipsec authentication method
  - ipsec authentication pre-shared-key
  - ipsec ike proposal
  - ipsec proposal
  - local-address
  - mtu
  - outside-address
  - remote-address
  - remote network
  - remote networks
  - username
  - wins-servers
- Настройка OpenVPN-сервера
  - address-range
  - authentication algorithm
  - bridge-group
  - crypto
  - client-isolation
  - client-max
  - compression
  - dns-server
  - duplicate-cn
  - encryption algorithm
  - ip address
  - login authentication
  - network
  - port
  - protocol
  - redirect-gateway
  - route
  - timers holdtime
  - timers keepalive
  - subnet
  - tunnel
  - username
  - wins-server
- Настройка WireGuard-сервера
  - dns-server

- [local-address](#)
- [mtu](#)
- [peer](#)
- [port](#)
- [private-key](#)
- [public-key](#)
- [subnet](#)

## Общие команды настройки удаленного доступа

### **authentication required disable**

Данной командой отключается обязательная аутентификация PPP-клиентов для PPTP или L2TP-сервера.

Использование отрицательной формы команды (no) включает обязательную аутентификацию PPP для PPTP- или L2TP-сервера.

### **Синтаксис**

```
[no] authentication required disable
```

### **Параметры**

Отсутствуют.

### **Значение по умолчанию**

Не задан.

### **Необходимый уровень привилегий**

15

### **Командный режим**

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

CONFIG-OPENVPN-SERVER

### **Пример**

```
esr(config-pptp-server)# authentication required disable
```

### **clear remote-access counters**

Данной командой осуществляется сброс счетчиков соединений OpenVPN, PPTP, Wireguard, L2TP over IPsec-пользователей.

### **Синтаксис**

```
clear remote-access counters [ pptp | l2tp | openvpn | wireguard ] [ server <SERVER-NAME> ] [ username <USER-NAME> ] [ ip-address <ADDR> ]
```

## Параметры

<SERVER-NAME> – имя профиля OpenVPN, PPTP, Wireguard или L2TP over IPsec-сервера;

<USER-NAME> – имя OpenVPN, PPTP или L2TP over IPsec-пользователя;

<ADDR> – IP-адрес OpenVPN, PPTP или L2TP over IPsec-пользователя.

При выполнении команды без параметра будут сброшены все счетчики соединений OpenVPN, PPTP, Wireguard и L2TP over IPsec-пользователей.

## Необходимый уровень привилегий

10

## Командный режим

ROOT

## Пример

```
esr# clear remote-access counters
```

## clear remote-access session

Данной командой осуществляется завершение соединений OpenVPN, PPTP и L2TP over IPsec-пользователей.

## Синтаксис

```
clear remote-access session [ pptp | l2tp | openvpn ] [ server <SERVER-NAME> ] [ username <USER-NAME> ] [ ip-address <ADDR> ]
```

## Параметры

<SERVER-NAME> – имя профиля OpenVPN, PPTP или L2TP over IPsec-сервера;

<USER-NAME> – имя OpenVPN, PPTP или L2TP over IPsec-пользователя;

<ADDR> – IP-адрес OpenVPN, PPTP или L2TP over IPsec-пользователя. При выполнении команды без параметра будут завершены все OpenVPN, PPTP и L2TP over IPsec-соединения.

## Необходимый уровень привилегий

10

## Командный режим

ROOT

## Пример

```
esr# clear remote-access session
```

## description

Команда используется для изменения описания профиля OpenVPN, PPTP и L2TP over IPsec-серверов.



Использование отрицательной формы команды (no) удаляет описание профиля.

### Синтаксис

```
description <DESCRIPTION>
```

```
no description
```

### Параметры

<DESCRIPTION> – описание профиля, задаётся строкой до 255 символов.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

CONFIG-OPENVPN-SERVER

CONFIG-WIREGUARD-SERVER

CONFIG-WIREGUARD-SERVER-PEER

### Пример

Установить описание для профиля PPTP-сервера:

```
esr(config-pptp-server)# description "Our remote workers"
```

### enable

Данной командой активируется конфигурируемый профиль серверов удаленного доступа.

Использование отрицательной формы команды (no) деактивирует конфигурируемый профиль.

### Синтаксис

```
[no] enable
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Выключено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

CONFIG-OPENVPN-SERVER  
 CONFIG-WIREGUARD-SERVER  
 CONFIG-WIREGUARD-SERVER-PEER

### Пример

```
esr(config-pptp-server)# enable
```

### encryption mppe

Данная команда включает шифрование MPPE (Microsoft Point-to-Point Encryption) для PPTP-соединений. Использование отрицательной формы команды (no) отключает шифрование.

### Синтаксис

```
[no] encryption mppe
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-PPTP-SERVER

### Пример

```
esr(config-pptp-server)# encryption mppe
```

### remote-access

Данной командой создается профиль сервера удаленного доступа. Использование отрицательной формы команды (no) удаляет указанный профиль.

### Синтаксис

```
[no] remote-access <SERVER-TYPE> <NAME>
```

### Параметры

<SERVER-TYPE> – тип сервера удаленного доступа. Может принимать значения l2tp, openvpn, pptp, wireguard.

<NAME> – имя профиля сервера удаленного доступа, задаётся строкой до 31 символа.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# remote-access l2tp remote-workers
esr(config-l2tp-server)#
```

**show remote-access configuration**

Командой выполняется просмотр параметров профилей OpenVPN, WireGuard PPTP и L2TP over IPsec-серверов.

**Синтаксис**

```
show remote-access configuration { pptp | l2tp | openvpn | wireguard} [ <NAME> ]
```

**Параметры**

<NAME> – имя профиля OpenVPN, WireGuard, PPTP или L2TP over IPsec сервера.

При выполнении команды без параметра будут показаны параметры всех OpenVPN, WireGuard, PPTP или L2TP over IPsec серверов.

**Необходимый уровень привилегий**

10

**Командный режим**

ROOT

**Пример**

```
esr# show remote-access configuration pptp pptp1
State:                Enabled
Description:          --
Security zone:        trusted
Authentication mode:  local
MTU:                  1500
Local address:        192.168.1.1
Remote address:       rem_pptp(10.0.10.20-10.0.10.40)
Outside address:      115.0.0.1
DNS server:           --
WINS server:          --
  Users
  ~~~~~
#    Name                State      Encrypted password
---  -
0    pptp                Enabled    8CB5107EA7005AFF
1    petr                 Enabled    CCE5513EE45A1EAC
```

## show remote-access counters

Командой выполняется просмотр счетчиков соединений OpenVPN, WireGuard, PPTP и L2TP over IPsec пользователей.

### Синтаксис

```
show remote-access counters [ pptp | l2tp | openvpn | wireguard] [ server <SERVER-NAME> ]
[ username <USER-NAME> ] [ ip-address <ADDR> ]
```

### Параметры

<SERVER-NAME> – имя профиля WireGuard, PPTP или L2TP over IPsec сервера;

<USER-NAME> – имя OpenVPN, PPTP или L2TP over IPsec пользователя;

<ADDR> – IP-адрес OpenVPN, PPTP или L2TP over IPsec пользователя;

При выполнении команды без параметра будут показаны счетчики всех соединений OpenVPN, WireGuard, PPTP и L2TP over IPsec пользователей.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример

```
esr# show remote-access counters
User          IP-address      UC recv      Bytes recv   Err recv     MC recv
-----
ivan          10.20.20.5      262          25365        0            0
fedor        20.20.20.160    59           5236         0            0
User          IP-address      UC sent      Bytes sent   Err sent
-----
ivan          10.20.20.5      249          29298        0
fedor        20.20.20.160    16           739          0
esr# show remote-access counters l2tp
PPTP Server: remote-workers
User: ivan(10.20.20.5)
Packets received:          231
Bytes received:            22229
Dropped on receive:        0
Receive errors:            0
Multicasts received:       0
Receive length errors:     0
Receive buffer overflow errors: 0
Receive CRC errors:        0
Receive frame errors:      0
Receive FIFO errors:       0
Receive missed errors:     0
Receive compressed:        0
Packets transmitted:       189
Bytes transmitted:         21858
Dropped on transmit:       0
Transmit errors:           0
Transmit aborted errors:    0
```

```

Transmit carrier errors:      0
Transmit FIFO errors:        0
Transmit heartbeat errors:    0
Transmit window errors:      0
Transmit compressed:         0
Collisions:                   0

```

## show remote-access status

Командой выполняется просмотр состояния соединений OpenVPN, WireGuard, PPTP и L2TP over IPsec пользователей.

### Синтаксис

```
show remote-access status [ pptp | l2tp | openvpn | wireguard] [ server <SERVER-NAME> ]
[ username <USER-NAME> ] [ ip-address <ADDR> ]
```

### Параметры

<SERVER-NAME> – имя профиля OpenVPN, WireGuard, PPTP или L2TP over IPsec сервера;

<USER-NAME> – имя OpenVPN, PPTP или L2TP over IPsec пользователя;

<ADDR> – IP-адрес OpenVPN, PPTP или L2TP over IPsec пользователя.

При выполнении команды без параметра будет показано состояние всех соединений OpenVPN, WireGuard, PPTP и L2TP over IPsec пользователей.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример

```

esr# show remote-access status
User           IP-address      Server
-----
ivan           10.20.20.5      pptp(remote-workers)
fedor          20.20.20.160    l2tp(remote-workers-l2tp)
Count sessions: 2

```

## Настройка L2TP over IPsec/PPTP-сервера

### authentication method

Данной командой разрешается использование метода аутентификации удаленных пользователей, подключающихся по протоколам PPTP или L2TP over IPsec.

Использование отрицательной формы команды (no) удаляет установленный режим.

### Синтаксис

```
[no] authentication method <METHOD>
```

## Параметры

<METHOD> – метод аутентификации, принимает значения [chap, mschap, mschap-v2, eap, pap].

## Значение по умолчанию

Разрешен только chap.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

## Пример

```
esr(config-pptp-server)# authentication method mschap
```

## authentication mode

Данной командой устанавливается режим аутентификации удаленных пользователей, подключающихся по протоколам PPTP или L2TP over IPsec.

Использование отрицательной формы команды (no) удаляет установленный режим.

## Синтаксис

```
authentication mode { local | radius }
```

```
no authentication mode
```

## Параметры

- local – режим аутентификации, использующий локальную базу пользователей конфигурируемого профиля.
- radius – режим, при котором аутентификация пользователей проходит через RADIUS-сервер.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

## Пример

```
esr(config-pptp-server)# authentication mode local
```

**dns-servers**

Данной командой указывается список DNS-серверов, которые будут использовать удаленные пользователи, подключающиеся по протоколам PPTP и L2TP over IPsec.

Использование отрицательной формы команды (no) удаляет настроенные адреса DNS-серверов.

**Синтаксис**

```
dns-servers object-group <NAME>
no dns-servers
```

**Параметры**

<NAME> – имя профиля IP-адресов, который содержит адреса необходимых DNS-серверов, задается строкой до 31 символа.

**Необходимый уровень привилегий**

10

**Командный режим**

```
CONFIG-PPTP-SERVER
CONFIG-L2TP-SERVER
```

**Пример**

```
esr(config-pptp-server)# dns-servers object-group pptp_dns
```

**dscp**

Команда задает значение кода DSCP для использования в IP-заголовке исходящих пакетов PPTP и L2TP over IPsec-серверов.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

**Синтаксис**

```
dscp <DSCP>
no dscp
```

**Параметры**

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

**Значение по умолчанию**

32

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

**Пример**

```
esr(config-pptp-server)# dscp 40
```

**ipsec authentication method**

Данной командой выбирается метод аутентификации по ключу для IKE-соединения. Аутентификация сообщений по ключу используется при установлении IKE-соединения, ключ задаётся командой «`ipsec authentication pre-shared-key`» (см раздел [ipsec authentication pre-shared-key](#)).

Использование отрицательной формы команды (`no`) устанавливает значение по умолчанию.

**Синтаксис**

```
ipsec authentication method pre-shared-key
```

```
no ipsec authentication method
```

**Параметры**

`pre-shared-key` – метод аутентификации, использующий предварительно полученные ключи шифрования.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-L2TP-SERVER

**Пример**

```
esr(config-l2tp-server)# ipsec authentication method pre-shared-key
```

**ipsec authentication pre-shared-key**

Данной командой устанавливается общий секретный ключ для аутентификации, который должен совпадать у обеих сторон, устанавливающих туннель.

Использование отрицательной формы команды (`no`) удаляет установленный ключ.

**Синтаксис**

```
ipsec authentication pre-shared-key { ascii-text { <TEXT> | encrypted <ENCRYPTED-TEXT> } |
hexadecimal {<HEX> | encrypted <ENCRYPTED-HEX> } }
```

```
no ipsec authentication pre-shared-key
```



**Параметры**

<TEXT> – строка [1..64] ASCII-символов.

<HEX> – число размером [1..32] байт задаётся строкой [2..128] символов в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

<ENCRYPTED-TEXT> – зашифрованный пароль размером [1..32] байт, задаётся строкой [2..128] символов;

<ENCRYPTED-HEX> – зашифрованное число размером [2..64] байт, задаётся строкой [2..256] символов.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-L2TP-SERVER

**Пример**

```
esr(config-l2tp-server)# ipsec authentication pre-shared-key ascii-text password
```

**ipsec ike proposal**

Данной командой для L2TP-сервера назначается шаблон, ограничивающий используемые методы аутентификации и шифрования протокола IKE.

Использование отрицательной формы команды (no) удаляет ограничения на использование методов аутентификации и шифрования протокола IKE.

**Синтаксис**

```
[no] ipsec ike proposal <NAME>
```

**Параметры**

<NAME> – имя ранее созданного профиля протокола IKE, задаётся строкой до 31 символа.

**Значение по умолчанию**

Ограничение отсутствует.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-L2TP-SERVER

## Пример

```
esr(config-l2tp-server)# ipsec ike proposal IKE_PROPOSAL
```

## ipsec proposal

Данной командой для L2TP-сервера назначается шаблон, ограничивающий используемые методы аутентификации и шифрования протокола IPsec.

Использование отрицательной формы команды (no) удаляет ограничения на использование методов аутентификации и шифрования протокола IPsec.

## Синтаксис

```
[no] ipsec proposal <NAME>
```

## Параметры

<NAME> – имя ранее созданного профиля IPsec, задаётся строкой до 31 символа.

## Значение по умолчанию

Ограничение отсутствует.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-L2TP-SERVER

## Пример

```
esr(config-l2tp-server)# ipsec proposal IPSEC_PROPOSAL
```

## local-address

Данной командой указывается IP-адрес, используемый PPTP или L2TP over IPsec сервером в качестве локального IP-адреса туннеля.

Использование отрицательной формы команды (no) удаляет настроенный локальный IP-адрес туннеля.

## Синтаксис

```
local-address { object-group <NAME> | ip-address <ADDR> }  
no local-address
```

## Параметры

<NAME> – имя профиля IP-адресов, который содержит локальный IP-адрес туннеля, задаётся строкой до 31 символа.

<ADDR> – локальный IP-адрес туннеля задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

**Пример**

```
esr(config-pptp-server)# local-address object-group pptp_local
```

**mtu**

Данной командой указывается MTU для интерфейсов, которые будут создаваться при подключении удаленных пользователей по протоколам PPTP и L2TP over IPsec.

Использование отрицательной формы команды (no) устанавливает значение MTU по умолчанию.

**Синтаксис**

mtu &lt;MTU&gt;

no mtu

**Параметры**

<MTU> – значение MTU, принимает значения в диапазоне [1280..1500].

**Значение по умолчанию**

1500

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

**Пример**

```
esr(config-pptp-server)# mtu 1400
```

**outside-address**

Данной командой указывается IP-адрес, который будет прослушиваться PPTP или L2TP over IPsec сервером на наличие входящих подключений.

Использование отрицательной формы команды (no) удаляет настроенный адрес для прослушивания.

## Синтаксис

```
outside-address { object-group <NAME> | ip-address <ADDR> | interface { <IF> | <TUN> } }
no outside-address
```

## Параметры

<NAME> – имя профиля IP-адресов, содержащий адрес, который будет прослушиваться PPTP или L2TP over IPsec сервером на наличие входящих подключений, задаётся строкой до 31 символа.

<ADDR> – IP-адрес, который будет прослушиваться PPTP или L2TP over IPsec сервером на наличие входящих подключений, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

<IF> – интерфейс, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#).

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

## Пример

```
esr(config-pptp-server)# outside-address object-group pptp_outside
```

## remote-address

Данной командой указывается список IP-адресов, из которого PPTP или L2TP over IPsec сервером выдаются динамические IP-адреса удаленным пользователям.

Использование отрицательной формы команды (no) удаляет список IP-адресов удаленных пользователей.

## Синтаксис

```
remote-address { object-group <NAME>| address-range <FROM-ADDR>-<TO-ADDR> }
no remote-address
```

## Параметры

<NAME> – имя профиля IP-адресов, который содержит список IP-адресов удаленных пользователей, задаётся строкой до 31 символа.

<FROM-ADDR> – начальный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

<TO-ADDR> – конечный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

**Пример**

```
esr(config-pptp-server)# remote-address object-group pptp_remote
```

**remote network**

Данной командой в таблицу маршрутизации добавляется информация о IP-подсети, которая будет доступна со стороны клиента при установлении динамического PPTP/L2TP-тунеля.

Использование отрицательной формы команды (no) удаляет IP-адрес подсети, доступной через динамический PPTP/L2TP-туннель.

**Синтаксис**

```
remote network <ADDR/LEN>
```

```
no remote network
```

**Параметры**

<ADDR/LEN> – IP-подсеть получателя. Параметр задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-PPP-USER

**Пример**

```
esr(config-ppp-user)# remote network 192.168.54.0/24
```

**remote networks**

Данной командой в таблицу маршрутизации добавляется информация о IP-подсетях, которые будут доступны со стороны клиента при установлении динамического PPTP/L2TP-тунеля.

Использование отрицательной формы команды (no) удаляет информацию о IP-подсетях, доступных через динамический PPTP/L2TP-туннель.

**Синтаксис**

```
remote networks <OBJ-GROUP-NETWORK-NAME>
```

```
no remote network
```

### Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IPv4/IPv6-адресов, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-PPP-USER

### Пример

```
esr(config-ppp-user)# remote network 192.168.54.0/24
```

### username

Данной командой создается пользователь для подключения к PPTP или L2TP over IPsec серверам. После выполнения данной команды маршрутизатор переходит в режим конфигурирования параметров PPP-пользователя.

Использование отрицательной формы команды (no) удаляет указанного пользователя.

Команда устанавливает режим командной строки PPTP USER или L2TP USER в зависимости от текущего командного режима.

### Синтаксис

```
[no] username <NAME>
```

### Параметры

<NAME> – имя пользователя, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

### Пример

```
esr(config-pptp-server)# username fedor  
esr(config-pptp-user)#
```

### wins-servers

Данной командой указывается список WINS-серверов, которые будут использовать удаленные пользователи, подключающиеся по протоколам PPTP и L2TP over IPsec.

Использование отрицательной формы команды (no) удаляет настроенные адреса WINS-серверов.

### Синтаксис

```
wins-servers object-group <NAME>
no wins-servers
```

### Параметры

<NAME> – имя профиля IP-адресов, который содержит адреса необходимых WINS-серверов, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

```
CONFIG-PPTP-SERVER
CONFIG-L2TP-SERVER
```

### Пример

```
esr(config-pptp-server)# wins-servers object-group l2tp_wins
```

## Настройка OpenVPN-сервера

### address-range

Данной командой указывается список IP-адресов, из которого OpenVPN-сервером выдаются динамические IP-адреса удаленным пользователям в режиме L2.

Использование отрицательной формы команды (no) удаляет список IP-адресов удаленных пользователей.

### Синтаксис

```
address-range <FROM-ADDR>-<TO-ADDR>
no address-range
```

### Параметры

<FROM-ADDR> – начальный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

<TO-ADDR> – конечный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

### Необходимый уровень привилегий

10

### Командный режим

```
CONFIG-OPENVPN-SERVER
```

## Пример

```
esr(config-openvpn-server)# address-range 192.168.1.10-192.168.1.250
```

## authentication algorithm

Данная команда определяет алгоритм аутентификации клиентов OpenVPN.

Использование отрицательной формы команды (no) устанавливает режим аутентификации по умолчанию.

## Синтаксис

```
authentication algorithm <ALGORITHM>
```

```
no authentication-algorithm
```

## Параметры

<ALGORITHM> – алгоритм аутентификации:

- 8-128 bits key size: md4, rsa-md4, md5, rsa-md5, mdc2, rsa-mdc2;
- 8-160 bits key size: sha, sha1, rsa-sha, rsa-sha1, rsa-sha1-2, dsa, dsa-sha, dsa-sha1, dsa-sha1-old, ripemd160, rsa-ripemd160, ecdsa-with-sha1;
- 8-224 bits key size: sha-224, rsa-sha-224;
- 8-256 bits key size: sha-256, rsa-sha-256;
- 8-384 bits key size: sha-384, rsa-sha-384;
- 8-512 bits key size: sha-512, rsa-sha-512, whirlpool.

## Значение по умолчанию

sha

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-OPENVPN-SERVER

## Пример

```
esr(config-openvpn-server)# authentication algorithm cleartext
```

## bridge-group

Данная команда используется для включения клиентских соединений по OpenVPN в L2-домен.

Использование отрицательной формы команды (no) исключает соединения из L2-домена.

## Синтаксис

```
bridge-group <BRIDGE-ID>
```

```
no bridge-group
```



## Параметры

<BRIDGE-ID> – идентификационный номер моста. Задается в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-OPENVPN-SERVER

## Пример

```
esr(config-openvpn-server)# bridge-group 15
```

## crypto

Данной командой указываются сертификаты и ключи. Сертификаты и ключи должны быть предварительно скопированы на маршрутизатор с помощью команды `copy`, описанной в разделе [copy](#).

Использование отрицательной формы команды (`no`) удаляет из профиля указанный сертификат.

## Синтаксис

```
crypto <CERTIFICATE-TYPE> <NAME>
```

```
no crypto <CERTIFICATE-TYPE>
```

## Параметры

<CERTIFICATE-TYPE> – тип сертификата или ключа, может принимать следующие значения:

- ca – сертификат удостоверяющего сервера;
- cert – сертификат OpenVPN-сервера;
- crl – список отозванных сертификатов;
- dh – ключ Диффи-Хеллмана;
- private-key – приватный ключ сервера;
- ta – HMAC-ключ.

<NAME> – имя сертификата или ключа, задаётся строкой до 31 символа.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-OPENVPN-SERVER

## Пример

```
esr(config-openvpn-server)# crypto ca ca.crt
```

**client-isolation**

Данной командой включается блокировка передачи данных между клиентами.  
Использование отрицательной формы команды (no) снимает блокировку.

**Синтаксис**

```
[no] client-isolation
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-OPENVPN-SERVER

**Пример**

```
esr(config-openvpn-server)# client-isolation
```

**client-max**

Данной командой устанавливается максимальное количество одновременных пользовательских сессий.  
Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
client-max <VALUE>
```

```
no client-max
```

**Параметры**

<VALUE> – максимальное количество пользователей, принимает значения [1..65535].

**Значение по умолчанию**

Не ограничено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-OPENVPN-SERVER

## Пример

```
esr(config-openvpn-server)# clients-max 500
```

## compression

Данной командой включается механизм сжатия передаваемых данных между клиентами и сервером OpenVPN.

Использование отрицательной формы команды (no) отключает механизм сжатия передаваемых данных.

## Синтаксис

```
[no] compression
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Выключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-OPENVPN-SERVER

## Пример

```
esr(config-openvpn-server)# compression
```

## dns-server

Данной командой указывается список DNS-серверов, которые будут использовать удаленные пользователи.

Использование отрицательной формы команды (no) удаляет настроенные адреса DNS-серверов.

## Синтаксис

```
dns-server <ADDR>
```

```
no dns-server { <ADDR> | all }
```

## Параметры

<ADDR> – IP-адрес DNS-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

all – удалить все сконфигурированные диапазоны IP-адресов.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-OPENVPN-SERVER

**Пример**

```
esr(config-openvpn-server)# dns-server 1.1.1.1
```

**duplicate-cn**

Данная команда разрешает подключение множества пользователей с одним сертификатом.

Использование отрицательной команды (no) запрещает использование одного сертификата более чем одному пользователю.

**Синтаксис**

```
[no] duplicate-cn
```

**Параметры**

Команда не имеет параметров.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-OPENVPN-SERVER

**Пример**

```
esr(config-openvpn-server)# duplicate-cn
```

**encryption algorithm**

Данной командой выбирается алгоритм шифрования, используемый при передачи данных.

Использование отрицательной формы команды (no) отключает шифрование.

**Синтаксис**

```
encryption algorithm <ALGORITHM>
```

```
no encryption algorithm
```

**Параметры**

<ALGORITHM> – идентификатор протокола шифрования, принимает значения: des, blowfish128, aes128, des-ede, aes192, 3des, desx, aes256.

**Значение по умолчанию**

Шифрование отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-OPENVPN-SERVER

**Пример**

```
esr(config-openvpn-server)# encryption algorithm aes128
```

**ip address**

Данная команда определяет статический IP-адрес для указанного клиента. Использование отрицательной формы команды (no) удаляет статический IP-адрес у клиента.

**Синтаксис**

```
[no] ip address <ADDR>
```

**Параметры**

<ADDR> – IP-адрес, имеет следующий формат:

AAA.BBB.CCC.DDD – IP-адрес подсети с маской в форме префикса, где AAA-DDD принимают значения [0..255].

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-OPENVPN-USER

**Пример**

```
esr(config-openvpn-server)# username client  
esr(config-openvpn-user)# ip address 10.10.100.15
```

**login authentication**

Данной командой осуществляется активация списка аутентификации пользователей для их авторизации.

В конфигурации по умолчанию активным является список с именем «default», данный список содержит один способ аутентификации – «local».

Использование отрицательной формы команды (no) деактивирует список аутентификации.

### Синтаксис

```
login authentication <NAME>  
no login authentication
```

### Параметры

<NAME> – имя списка, задаётся строкой до 31 символа.

### Значение по умолчанию

default

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-OPENVPN-SERVER

### Пример

```
esr(config-openvpn-server)# login authentication OPENVPN-LIST
```

### network

Данной командой определяется подсеть, из которой выдаются IP-адреса пользователям. Первый IP-адрес в подсети выступает шлюзом для пользовательских сессий.

Использование отрицательной формы команды (no) удаляет данную подсеть.

### Синтаксис

```
network <ADDR/LEN>  
no network
```

### Параметры

<ADDR/LEN> – IP-подсеть подсети, имеет следующий формат:

AAA.BBB.CCC.DDD/EE – IP-адрес подсети с маской в форме префикса, где AAA-DDD принимают значения [0..255] и EE принимает значения [16..29].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-OPENVPN-SERVER

### Пример

```
esr(config-openvpn-server)# network 192.168.25.0/24
```

## port

Данной командой устанавливается TCP/UDP-порт, который будет прослушиваться OpenVPN-сервером. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
port <PORT>  
no port
```

### Параметры

<PORT> – TCP/UDP-порт, принимает значения [1..65535].

### Значение по умолчанию

1194

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-OPENVPN-SERVER

### Пример

```
esr(config-openvpn-server)# port 5000
```

## protocol

Данной командой устанавливается инкапсулирующий протокол. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
protocol <PROTOCOL>  
no protocol
```

### Параметры

<TYPE> – тип инкапсуляции, возможные значения:

- TCP-инкапсуляция в TCP-сегменты;
- UDP-инкапсуляция в UDP-дейтаграммы.

### Значение по умолчанию

Остановлено.

### Необходимый уровень привилегий

15

## Командный режим

CONFIG-OPENVPN-SERVER

### Пример

```
esr(config-openvpn-server)# protocol udp
```

## redirect-gateway

Данная команда включает анонсирование маршрута по умолчанию для OpenVPN-соединений, что приводит к замене маршрута по умолчанию на клиентской стороне. Новым шлюзом по умолчанию станет IP-адрес OpenVPN-сервера.

Использование отрицательной формы команды (no) отключает анонсирование маршрута по умолчанию.

### Синтаксис

```
[no] redirect-gateway
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-OPENVPN-SERVER

### Пример

```
esr(config-openvpn-server)# redirect-gateway
```

## route

Данной командой включается анонсирование указанных подсетей, шлюзом является IP-адрес OpenVPN-сервера (первый IP-адрес из подсети, заданной с помощью команды `network`, описанной в разделе [network](#)).

Использование отрицательной формы команды (no) отключает анонсирование указанных подсетей.

### Синтаксис

```
route <ADDR/LEN>
```

```
no route { <ADDR/LEN> | all }
```

### Параметры

<ADDR/LEN> – IP-подсеть, имеет следующий формат:

AAA.BBB.CCC.DDD/EE – IP-адрес подсети с маской в форме префикса, где AAA-DDD принимают значения [0..255] и EE принимает значения [1..32].



**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-OPENVPN-SERVER

**Пример**

```
esr(config-openvpn-server)# route 192.168.25.0/24, 192.168.26.0/24
```

**timers holdtime**

Данной командой устанавливается временной интервал, по истечении которого встречная сторона считается недоступной. Таймер запускается после установления отношений соседства и начинает отсчёт от 0. Таймер сбрасывается при получении каждого ответа на *keepalive*-сообщение от встречной стороны. Рекомендуется устанавливать значение таймера не менее  $3 * \textit{keepalive}$ .

Использование отрицательной формы команды (*no*) устанавливает значение по умолчанию.

**Синтаксис**

```
timers holdtime <TIME>
```

```
no timers holdtime
```

**Параметры**

<TIME> – время в секундах, принимает значения [2..65535].

**Значение по умолчанию**

120

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-OPENVPN-SERVER

**Пример**

```
esr(config-openvpn-server)# timers holdtime 360
```

**timers keepalive**

Данной командой устанавливается временной интервал, по истечении которого идет проверка соединения со встречной стороной.

Использование отрицательной формы команды (*no*) устанавливает значение по умолчанию.

**Синтаксис**

```
timers keepalive <TIME>
no timers keepalive
```

**Параметры**

<TIME> – время в секундах, принимает значения [1..32767].

**Значение по умолчанию**

10

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-OPENVPN-SERVER

**Пример**

```
esr(config-openvpn-server)# timers keepalive 120
```

**subnet**

Данная команда определяет подсеть, которая должна быть доступна через подключение указанного пользователя OpenVPN-сервера. После подключения пользователя к OpenVPN-серверу в таблице маршрутизации появляется маршрут в указанную подсеть через динамически созданный туннель.

Использование отрицательной формы команды (no) удаляет подсеть, работающую за данным пользователем.

**Синтаксис**

```
[no] subnet <ADDRLEN>
```

**Параметры**

<ADDR/LEN> – IP-подсеть, имеет следующий формат:

AAA.BBB.CCC.DDD/EE – IP-адрес подсети с маской в форме префикса, где AAA-DDD принимают значения [0..255] и EE принимает значения [16..32].

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-OPENVPN-USER

**Пример**

```
esr(config-openvpn-server)# username client
```

```
esr(config-openvpn-user)# subnet 192.168.25.128/28
```

## tunnel

Данной командой определяется тип соединения с частной сетью через OpenVPN-сервер. Использование отрицательной формы команды (no) удаляет текущее значение.

### Синтаксис

```
tunnel <TYPE>  
no tunnel
```

### Параметры

<TYPE> – инкапсулирующий протокол, принимает значения:

- ip – соединение точка-точка;
- ethernet – подключение к L2-домену.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-OPENVPN-SERVER

### Пример

```
esr(config-openvpn-server)# tunnel ip
```

## username

Данная команда позволяет перейти в режим конфигурирования для указанного пользователя OpenVPN-сервера.

Использование отрицательной формы команды (no) восстанавливает настройки пользователя по умолчанию.

### Синтаксис

```
[no] username { <NAME> | all }
```

### Параметры

<NAME> – имя пользователя, задаётся строкой до 31 символа;

all – ключ, используемый для удаления всех ранее созданных пользователей.

### Необходимый уровень привилегий

15

**Командный режим**

CONFIG-OPENVPN-SERVER

**Пример**

```
esr(config-openvpn-server)# username client
esr(config-openvpn-user)#
```

**wins-server**

Данной командой указывается список WINS-серверов, которые будут использовать удаленные пользователи.

Использование отрицательной формы команды (no) удаляет настроенные адреса WINS-серверов.

**Синтаксис**

```
wins-server <ADDR>
no wins-server { <ADDR> | all }
```

**Параметры**

<ADDR> – IP-адрес WINS-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

all – удалить все сконфигурированные IP-адреса DNS-серверов.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-OPENVPN-SERVER

**Пример**

```
esr(config-openvpn-server)# wins-servers 1.1.1.1
```

**Настройка WireGuard-сервера****dns-server**

Данной командой указывается список DNS-серверов, которые будут использовать удаленные пользователи.

Использование отрицательной формы команды (no) удаляет настроенные адреса DNS-серверов.

**Синтаксис**

```
dns-server { <ADDR> | <ADDR>,<ADDR> }
no dns-server { <ADDR> | all }
```

## Параметры

<ADDR> – IP-адрес DNS-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

all – удалить все сконфигурированные диапазоны IP-адресов.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WIREGUARD-SERVER

## Пример

```
esr(config-wireguard-server)# dns-server 8.8.8.8, 8.8.4.4
```

## local-address

Данной командой устанавливается статический IP-адрес конфигурируемого сервера.

Использование отрицательной формы команды (no) удаляет установленный IP-адрес.

## Синтаксис

local-address <ADDR/LEN>

no local-address

## Параметры

<ADDR/LEN> – IP-адрес и длина маски подсети, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WIREGUARD-SERVER

## Пример

```
esr(config-wireguard-server)# local-address 10.10.0.1/24
```

## mtu

Данной командой указывается MTU для интерфейсов, которые будут создаваться при подключении удаленных пользователей.

Использование отрицательной формы команды (no) устанавливает значение MTU по умолчанию.

**Синтаксис**

mtu <MTU>

no mtu

**Параметры**

<MTU> – 552–10000.

**Значение по умолчанию**

1500

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WIREGUARD-SERVER

**Пример**

```
esr(config-wireguard-server)# mtu 1420
```

**peer**

Данная команда используется для перехода к настройкам разрешённых туннелей. Использование отрицательной формы команды (no) удаляет разрешённый туннель.

**Синтаксис**

[no] peer <COUNT>

**Параметры**

<COUNT> – номер соответствующего пира, принимает значения [1..16].

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-WIREGUARD-SERVER

**Пример**

```
esr(config-wireguard-server)# peer 1
```

**port**

Данной командой устанавливается UDP-порт, который будет прослушиваться WireGuard-сервером.

**Синтаксис**

port <PORT>

no port

**Параметры**

<PORT> – UDP-порт, принимает значения [1..65535].

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-WIREGUARD-SERVER

**Пример**

```
esr(config-wireguard-server)# port 43021
```

**private-key**

Данной командой указывается необходимый приватный ключ WireGuard-сервера.

**Синтаксис**

private-key <NAME>

no private-key

**Параметры**

<NAME> – имя приватного ключа, задаётся строкой до 31 символа.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-WIREGUARD-SERVER

**Пример**

```
esr(config-wireguard-server)# private-key server.priv
```

**public-key**

Данной командой указывается необходимый публичный ключ клиента WireGuard.

**Синтаксис**

public-key <NAME>

no public-key

### Параметры

<NAME> – имя приватного ключа, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-WIREGUARD-SERVER-PEER

### Пример

```
esr(config-wireguard-server-peer)# public-key client.pub
```

### subnet

Данной командой указывается список IP-адресов, которым будет разрешено находиться внутри туннеля.

### Синтаксис

```
subnet <OBJ-GROUP-NETWORK-NAME>
```

```
no subnet
```

### Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, содержащего префиксы подсетей назначения, задается строкой до 31 символа.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-WIREGUARD-SERVER-PEER

### Пример

```
esr(config-wireguard-server-peer)# subnet client
```



## 23 Маршрутизация

- Общие настройки маршрутизации
- Общие команды анонсирования и приема маршрутов
- Маршрутизация на основе политик (PBR)
- Настройка связок ключей
- Настройка параметров протокола BFD
- Настройка статических маршрутов IPv4/IPv6
- Настройка протокола BGP
- Настройка протоколов RIP и RIPNG
- Настройка протоколов OSPF и OSPFv3
- Настройка протокола IS-IS

### Общие настройки маршрутизации

- authentication algorithm
- authentication key
- enable
- graceful-restart
- graceful-restart timeout
- ip path-mtu-discovery age-timer
- ip path-mtu-discovery disable
- ip path-mtu-discovery min-mtu
- ip path-mtu-discovery mode
- ip protocols max-routes
- ip protocols preference
- ipv6 protocols max-routes
- ipv6 tcp adjust-mss
- show ip protocols
- show ip route
- show ipv6 protocols
- show ipv6 route

#### authentication algorithm

Данная команда определяет алгоритм аутентификации.

Использование отрицательной формы команды (no) отключает аутентификацию.

#### Синтаксис

```
authentication algorithm <ALGORITHM>
```

```
no authentication algorithm
```

#### Параметры

<ALGORITHM> – алгоритм аутентификации:

- cleartext – пароль, передается открытым текстом (доступно только для режимов CONFIG-RIP и CONFIG-OSPF-VLINK);
- md5 – пароль хешируется по алгоритму md5.

#### Необходимый уровень привилегий

10

**Командный режим**

CONFIG-RIP

CONFIG-BGP

CONFIG-BGP-VRF

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

CONFIG-OSPF-VLINK

**Пример**

```
esr(config-rip)# authentication algorithm cleartext
```

**authentication key**

Данная команда устанавливает пароль для аутентификации с соседом.

Использование отрицательной формы команды (no) удаляет пароль.

**Синтаксис**

```
authentication key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
```

```
no authentication key
```

**Параметры**

<CLEAR-TEXT> – пароль, задаётся строкой от 8 до 16 символов.

<ENCRYPTED-TEXT> – зашифрованный пароль размером 8-16 байт (16-32 символа) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-RIP

CONFIG-BGP

CONFIG-BGP-VRF

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

CONFIG-OSPF-VLINK

## Пример

```
esr(config-bgp-af)# authentication key ascii-text 123456789  
esr(config-bgp-af)# authentication key ascii-text encrypted CDE65039E5591FA3F1
```

## enable

Данной командой включается конфигурируемый протокол маршрутизации, область, виртуальное соединение, соседство.

Использование отрицательной формы команды (no) отключает RIP-протокол.

## Синтаксис

```
[no] enable
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Отключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-RIP

CONFIG-RIPNG

CONFIG-OSPF

CONFIG-OSPF-AREA

CONFIG-OSPF-VLINK

CONFIG-IPV6-OSPF

CONFIG-IPV6-OSPF-AREA

CONFIG-IPV6-OSPF-VLINK

CONFIG-ISIS

CONFIG-BGP

CONFIG-BGP-VRF

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

CONFIG-BGP-NEIGHBOR-FAMILY

CONFIG-BGP-VRF-NEIGHBOR-FAMILY

CONFIG-BGP-AGGREGATE

CONFIG-BGP-VRF-AGGREGATE

**Пример 1**

```
esr(config-rip)# enable
```

**Пример 2**

```
esr(config-ospf)# enable
```

**Пример 3**

```
esr(config-isis)# enable
```

**Пример 4**

```
esr(config-bgp-neighbor)# enable
```

**graceful-restart**

Данной командой включается механизм поддержания активного состояния соседства с сохранением маршрутной информации на момент перезапуска протокола маршрутизации.

Использование отрицательной формы команды (no) отключает данный механизм.

**Синтаксис**

```
[no] graceful-restart
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

CONFIG-OSPF

## CONFIG-IPV6-OSPF

### Пример

```
esr(config-bgp-neighbor)# graceful-restart
```

### graceful-restart timeout

Данной командой устанавливается временной интервал, в течении которого следует поддерживать активное состояние соседства на момент перезапуска протокола маршрутизации. По истечении временного интервала встречная сторона считается недоступной. Используется совместно с командой [graceful-restart](#).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
graceful-restart timeout <TIME>  
no graceful-restart timeout
```

### Параметры

<TIME> – время в секундах, принимает значения [1..65535].

### Значение по умолчанию

120

### Необходимый уровень привилегий

10

### Командный режим

```
CONFIG-BGP-GROUP  
CONFIG-BGP-VRF-GROUP  
CONFIG-BGP-NEIGHBOR  
CONFIG-BGP-VRF-NEIGHBOR  
CONFIG-OSPF  
CONFIG-IPV6-OSPF
```

### Пример

```
esr(config-bgp-neighbor)# graceful-restart timeout 60
```

### ip path-mtu-discovery age-timer

Данной командой настраивается временной интервал, в течение которого поддерживается текущий режим работы PMTU. По истечении данного времени режим PMTU сменится на тот, который указан в конфигурации.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ip path-mtu-discovery age-timer <TIME>
```

```
no ip path-mtu-discovery age-timer
```

### Параметры

<TIME> – время в минутах, принимает значения [1..30];

### Значение по умолчанию

10

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# ip path-mtu-discovery age-timer 15
```

### ip path-mtu-discovery disable

Данная команда запрещает/разрешает поиск PMTU для протоколов TCP, SCTP, DCCP.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ip path-mtu-discovery disable
```

```
no ip path-mtu-discovery disable
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Поиск PMTU разрешен.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

## Пример

```
esr(config)# ip path-mtu-discovery disable
```

### **ip path-mtu-discovery min-mtu**

Данной командой настраивается минимальное значение MTU для использования в PMTU.  
Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### **Синтаксис**

```
ip path-mtu-discovery min-mtu <VALUE>  
no ip path-mtu-discovery min-mtu
```

### **Параметры**

<VALUE> – длина в байтах, принимает значения [64..10000].

### **Значение по умолчанию**

552

### **Необходимый уровень привилегий**

10

### **Командный режим**

CONFIG

## Пример

```
esr(config)# ip path-mtu-discovery min-mtu 1000
```

### **ip path-mtu-discovery mode**

Данной командой настраивается режим работы PMTU.  
Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### **Синтаксис**

```
ip path-mtu-discovery mode <MODE>  
no ip path-mtu-discovery mode
```

### **Параметры**

<MODE> – режим работы PMTU:

- default – режим работы по умолчанию, в соответствии RFC1191;
- icmp-discard – игнорировать входящие PMTU-сообщения;
- secure – выполнять фрагментацию только для протоколов TCP, SCTP and DCCP.

### Значение по умолчанию

default

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# ip path-mtu-discovery mode icmp-discard
```

### ip protocols max-routes

Данная команда позволяет настроить емкость таблиц маршрутизации.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ip protocols <PROTOCOL> max-routes <VALUE>
```

```
no ip protocols <PROTOCOL> max-routes
```

### Параметры

<PROTOCOL> – вид протокола, принимает значения: rip (только в глобальном режиме), ospf, isis, bgp;

<VALUE> – количество маршрутов в маршрутной таблице, принимает значения в диапазоне:

- BGP
  - ESR-1000/1200/1500/1511/1700/3100/3200/3200L/3300 [0..5000000];
  - ESR-20/21/30/31/100/200 [0..2500000],
  - ESR-10/12V/12VF/15/15R/15VF [0..1000000].
- OSPF и IS-IS
  - ESR-1000/1200/1500/1511/1700/3100/3200/3200L/3300 [0..500000];
  - ESR-20/21/30/31/100/200 [0..300000];
  - ESR-10/12V/12VF/15/15R/15VF [0..30000].
- RIP
  - ESR-20/21/30/100/200/1000/1200/1500/1511/1700/3100/3200/3200L/3300 [0..10000];
  - ESR-10/12V/12VF/15/15R/15VF [0..1000].

### Значение по умолчанию для глобального режима

BGP

- ESR-1000/1200/1500/1511/1700/3100/3200/3200L/3300 (5000000);
- ESR-20/21/30/31/100/200 (2500000);
- ESR-10/12V/12VF/15/15R/15VF (1000000).



**OSPF и IS-IS**

- ESR-1000/1200/1500/1511/1700/3100/3200/3200L/3300 (500000);
- ESR-20/21/30/31/100/200 (300000);
- ESR-10/12V/12VF/15/15R/15VF (30000).

**RIP**

- ESR-20/21/30/100/200/1000/1200/1500/1511/1700/3100/3200/3200L/3300 (10000);
- ESR-10/12V/12VF/15/15R/15VF (1000).

**Значение по умолчанию для vrf**

0

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

CONFIG-VRF

**Пример**

```
esr(config)# ip protocols ospf max-routes 4400
```

**ip protocols preference**

Данная команда позволяет настроить приоритетность протоколов маршрутизации для основной таблицы маршрутизации.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
ip protocols <PROTOCOL> preference <VALUE>
```

```
no ip protocols <PROTOCOL> preference
```

**Параметры**

<PROTOCOL> – вид протокола, принимает значения: static, rip, ospf, isis, bgp, dhcp, l2tp, pppoe, pptp, nhrp;

<VALUE> – приоритетность протокола, принимает значения в диапазоне [1..255].

**Значение по умолчанию**

BGP (170)

IS-IS (160)

OSPF (150)

RIP (100)

DHCP (40)

L2TP (50)  
 PPPoE (50)  
 PPTP (50)  
 NHRP (20)  
 Static (1)

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# ip protocols ospf preference 44
```

### ipv6 protocols max-routes

Данная команда позволяет настроить емкость таблиц маршрутизации IPv6.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ipv6 protocols <PROTOCOL> max-routes <VALUE>
```

```
no ipv6 protocols <PROTOCOL> max-routes
```

### Параметры

<PROTOCOL> – вид протокола, принимает значения: rip (только в глобальном режиме), ospf, isis, bgp;

<VALUE> – количество маршрутов в маршрутной таблице, принимает значения в диапазоне:

- BGP
  - ESR-1000/1200/1500/1511/1700/3100/3200/3200L/3300 [0..500000];
  - ESR-20/21/30/31/100/200 [0..250000];
  - ESR-10/12V/12VF/15/15R/15VF [0.. 100000].
- OSPFv3 и IS-IS
  - ESR-1000/1200/1500/1511/1700/3100/3200/3200L/3300 [0..500000];
  - ESR-20/21/30/31/100/200 [0..300000];
  - ESR-10/12V/12VF/15/15R/15VF [0..30000].
- IPv6 BGP
  - ESR-1000/1200/1500/1511/1700/3100/3200/3200L/3300 [0..500000];
  - ESR-1000/1200/1500 [0..300000];
  - ESR-10/12V/12VF/15/15R/15VF/20/21/30/31/100/200 [0..150000].

### Значение по умолчанию для глобального режима

OSPFv3

- ESR-1000/1200/1500/1511/1700/3100/3200/3200L/3300 (500000);
- ESR-20/21/30/31/100/200 (300000);

- ESR-10/12V/12VF/15/15R/15VF [1..30000].

#### IPv6 BGP

- ESR-1000/1200/1500/1511/1700/3100/3200/3200L/3300 (5000000);
- ESR-20/21/30/31/100/200 [1..2500000];
- ESR-10/12V/12VF/15/15R/15VF [1..100000].

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG

CONFIG-VRF

#### Пример

```
esr(config)# ipv6 protocols ospf max-routes 4400
```

#### ipv6 tcp adjust-mss

Данной командой переопределяется значение поля MSS (Maximum segment size) во входящих TCP-пакетах.

Использование отрицательной формы команды (no) отключает корректировку значение поля MSS.

#### Синтаксис

```
ipv6 tcp adjust-mss <MSS>
```

```
no ipv6 tcp adjust-mss
```

#### Параметры

<MSS> – значение MSS, принимает значения в диапазоне [40..1940].

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-IF-IG

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

## Пример

```
esr(config-if-gi)# ipv6 tcp adjust-mss 1400
```

## show ip protocols

Данная команда выводит информацию о настройках протоколов IP-маршрутизации.

## Синтаксис

```
show ip protocols [ <PROTOCOL> ] [ vrf <VRF> ]
```

## Параметры

<PROTOCOL> – протокол маршрутизации, по которому должна быть отображена информация:

- bgp;
- ospf;
- isis;
- rip;
- static;
- dhcp;
- pptp;
- pppoe;
- l2tp.

Без указания протокола маршрутизации, выводится информация о настройках всех протоколов маршрутизации.

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
esr# sh ip protocols
BGP:
  Max routes:  --
  Preference:  170
OSPF:
  Max routes:  --
  Preference:  150
RIP:
  Max routes:  --
  Preference:  100
Static:
```

Preference: 1

## show ip route

Данная команда позволяет просмотреть таблицу маршрутизации устройства. Если задан параметр <SUBNET>, то детально отображаются маршруты к данной подсети. Если задан параметр <VRF>, то команда отображает таблицу маршрутизации указанного экземпляра VRF.

### Синтаксис

```
show ip route [ vrf <VRF> ] [ { <SUBNET> [ long-prefix ] | all | summary | arp-proxy | <PROTOCOL> } ]
```

### Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа;

<SUBNET> – адрес назначения, опциональный параметр, может быть задан в следующих видах:

- AAA.BBB.CCC.DDD – IP-адрес хоста, где каждая часть принимает значения [0..255];
- AAA.BBB.CCC.DDD/NN – IP-адрес подсети с маской в виде префикса, где AAA-DDD принимают значения [0..255] и NN принимает значения [1..32];
- all – выводит информацию о всех маршрутах, включая не выбранные для FIB;
- long-prefix – выводит информацию о маршрутах до сетей являющихся подсетью заданной;
- summary – выводит суммарную статистику протоколов маршрутизации;
- arp-proxy – выводит информацию о настроенных пулах arp-nat-proxy на интерфейсах и туннелях;
- <PROTOCOL> – фильтрация по типу протоколу (bgp, connected, ospf, isis, rip, static, nhrp).

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
esr# show ip route
Codes: C - connected, S - static, R - RIP derived,
       O - OSPF derived, IA - OSPF inter area route,
       E1 - OSPF external type 1 route, E2 - OSPF external type 2 route
       B - BGP derived, D - DHCP derived, K - kernel route, V - VRRP route
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       H - NHRP, * - FIB route

C      * 192.168.1.0/24    [0/0]   dev br1                [direct 01:14:16]
C      * 10.100.100.0/24  [0/0]   dev gi1/0/5           [direct 01:14:17]
esr# show ip route summary
Direct Connected: 12
Static:           46
RIP:              0
OSPF:             2000
BGP:              1000000
DHCP:             1
IS-IS:            0
NHRP:             12
```

## show ipv6 protocols

Данная команда выводит информацию о настройках протоколов IPv6-маршрутизации.

### Синтаксис

```
show ipv6 protocols [ <PROTOCOL> ] [ vrf <VRF> ]
```

### Параметры

<PROTOCOL> – протокол маршрутизации, по которому должна быть отображена информация:

- bgp;
- ospf;
- isis;
- static;
- dhcp;
- pppoe;
- pptp;
- l2tp.

Без указания протокола маршрутизации выводится информация о настройках всех протоколов маршрутизации.

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
esr# sh ipv6 protocols
BGP:
  Max routes:  --
  Preference:  170
OSPF:
  Max routes:  --
  Preference:  150
Static:
  Preference:  1
```

## show ipv6 route

Команда для просмотра таблицы маршрутизации устройства. Если задан параметр <SUBNET>, то детально отображаются маршруты к данной подсети. Если задан параметр <VRF>, то команда отображает таблицу маршрутизации указанного экземпляра VRF.

### Синтаксис

```
show ipv6 route [ vrf <VRF> ] [ { <SUBNET> | all | summary | <PROTOCOL> } ]
```

## Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа;

<SUBNET> – адрес назначения, опциональный параметр, может быть задан в следующих видах:

- X:X:X:X – IPv6-адрес хоста, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF];
- X:X:X:X::X/EE – IPv6-адрес подсети с маской в виде префикса, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128];
- all – выводит информацию о всех маршрутах, включая неактивные;
- summary – выводит суммарную статистику протоколов маршрутизации;
- <PROTOCOL> – фильтрация по типу протоколу (bgp, connected, ospf, isis, static).

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```

esr# show ipv6 route
Codes: C – connected, S – static, R – RIP derived,
       O – OSPF derived, IA – OSPF inter area route,
       E1 – OSPF external type 1 route, E2 – OSPF external type 2 route
       B – BGP derived, D – DHCP derived, K – kernel route,
       * – FIB route
S      * ::/0                [1/0]   via fc00::1 on gi1/0/5           [static 03:16:23]
S      * 2001::/120         [1/6]   dev gi1/0/5                     [static 03:16:23]
C      * fc00::/120        [0/0]   dev gi1/0/5                     [direct 03:16:23]
S      * fc00:3::1/128     [1/0]   via fc00::1 on gi1/0/5           [static 03:16:23]
esr# show ipv6 route summary
Direct Connected: 1
Static:           3
RIP:              0
OSPF:             0
BGP:              0

```

## Общие команды анонсирования и приема маршрутов

- [default-information-originate](#)
- [description](#)
- [ip prefix-list](#)
- [ipv6 prefix-list](#)
- [network](#)
- [permit/deny](#)
- [prefix-list](#)
- [redistribute bgp](#)
- [redistribute connected](#)
- [redistribute ipv6 bgp](#)
- [redistribute ipv6 ospf](#)
- [redistribute isis](#)
- [redistribute ospf](#)
- [redistribute rip](#)
- [redistribute static](#)

### default-information-originate

Данной командой включается генерация и отправка маршрута по умолчанию, если он прописан в таблице маршрутизации FIB, для:

- NSSA-области (анонсирование маршрута) в качестве NSSA-LSA;
- BGR IPv4/IPv6 address family.

Использование отрицательной формы команды (no) отключает генерацию маршрута по умолчанию.

### Синтаксис

```
[no] default-information-originate
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-BGP

CONFIG-BGP-VRF

CONFIG-OSPF-AREA

CONFIG-IPV6-OSPF-AREA

### Пример

```
esr(config-ospf-area)# default-information-originate
```



## description

Данная команда используется для изменения описания конфигурируемого списка IP-подсетей или IPv6-подсетей.

Использование отрицательной формы команды (no) удаляет установленное описание.

## Синтаксис

```
description <DESCRIPTION>  
no description
```

## Параметры

<DESCRIPTION> – описание списка IP-подсетей или IPv6-подсетей, задаётся строкой до 255 символов.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-PL  
CONFIG-IPV6-PL

## Пример

```
esr(config-route-map-rule)# description "Drop Local NETs"
```

## ip prefix-list

Данной командой создается список IP-подсетей, который в дальнейшем будет использоваться для фильтрации анонсируемых и получаемых IP-маршрутов.

Использование отрицательной формы команды (no) удаляет список префиксов.

## Синтаксис

```
[no] ip prefix-list <NAME>
```

## Параметры

<NAME> – имя конфигурируемого списка подсетей, задаётся строкой до 31 символа.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# ip prefix-list ospf_in
```

## ipv6 prefix-list

Данной командой создается список IPv6-подсетей, который в дальнейшем будет использоваться для фильтрации анонсируемых и получаемых IPv6-маршрутов.

Использование отрицательной формы команды (no) удаляет список префиксов.

## Синтаксис

```
[no] ipv6 prefix-list <NAME>
```

## Параметры

<NAME> – имя конфигурируемого списка подсетей, задаётся строкой до 31 символа.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# ipv6 prefix-list ospfv3_in
```

## network

Данной командой включается анонсирование указанной подсети.

При использовании в CONFIG-OSPF-AREA и CONFIG-IPV6-OSPF-AREA подсеть анонсируется как External Type 2 (LSA5).

Использование отрицательной формы команды (no) отключает анонсирование указанной подсети.

## Синтаксис

```
[no] network <ADDR/LEN> [route-map <NAME>]
```

## Параметры

<ADDR/LEN> – IP-подсеть, имеет один из следующих форматов:

- AAA.BBB.CCC.DDD/EE – IP-адрес подсети с маской в форме префикса, где AAA-DDD принимают значения [0..255] и EE принимает значения [1..32];
- X:X:X::X/EE – IPv6-адрес и маска подсети, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128].

<NAME> – имя маршрутной карты, которая в дальнейшем будет использоваться для фильтрации и модификации анонсируемых BGP-маршрутов, задаётся строкой до 31 символа;

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-RIP

CONFIG-RIPNG

CONFIG-OSPF-AREA

CONFIG-IPV6-OSPF-AREA

CONFIG-BGP-FAMILY

CONFIG-BGP-VRF-FAMILY

### Пример

```
esr(config-bgp)# network 192.168.25.0/24
```

### permit/deny

Данной командой разрешаются (permit) или запрещаются (deny) списки префиксов.

### Синтаксис

```
permit [ { object-group <OBJ-GROUP-NETWORK-NAME> | <ADDR/LEN> | <IPV6-ADDR/LEN> } ]  
[ { eq <LEN> | le <LEN> | ge <LEN> [ le <LEN> ] } ]
```

```
deny [ { object-group <OBJ-GROUP-NETWORK-NAME> | <ADDR/LEN > | <IPV6-ADDR/LEN> } ] [ { eq  
<LEN> | le <LEN> | ge <LEN> [ le <LEN> ] } ]
```

```
no { object-group <OBJ-GROUP-NETWORK-NAME> | prefix { <ADDR/LEN> | <IPV6-ADDR/LEN> } }
```

### Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP/IPv6-адресов, задаётся строкой до 31 символа;

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<LEN> – длина префикса, принимает значения [1..32] в IP-списках префиксов и [1..128] в IPv6-списках префиксов;

eq – при указании команды длина префикса должна соответствовать указанной;

le – при указании команды длина префикса должна быть меньше либо соответствовать указанной;

ge – при указании команды длина префикса должна быть больше либо соответствовать указанной.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-PL

CONFIG-IPV6-PL

## Пример

```
esr(config-pl)# permit static ge 24 le 28
```

## prefix-list

Данной командой добавляется фильтрация подсетей во входящих или исходящих обновлениях. Использование отрицательной формы команды (no) отключает фильтрацию.

## Синтаксис

```
prefix-list <PREFIX-LIST-NAME> { in | out }  
no prefix-list { in| out }
```

## Параметры

<PREFIX-LIST-NAME> – имя сконфигурированного списка подсетей, задаётся строкой до 31 символа:

in – фильтрация входящих маршрутов;

out – фильтрация анонсируемых маршрутов.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR-FAMILY

CONFIG-BGP-VRF-NEIGHBOR-FAMILY

CONFIG-OSPF

CONFIG-IPV6-OSPF

CONFIG-RIP

CONFIG-RIPNG

## Пример

```
esr(config-rip)# prefix-list rip_in in
```

## redistribute bgp

Данной командой включается анонсирование маршрутов автономной системы BGP.

Использование отрицательной формы команды (no) отключает анонсирование маршрутов автономной системы BGP.

### Синтаксис

```
redistribute bgp <AS> [metric <TYPE> <METRIC>] [ route-map <NAME> ] [ <LEVEL> ]
no redistribute bgp <AS>
```

### Параметры

<AS> – номер автономной системы, из которой будут анонсироваться маршруты, может принимать значения [1..4294967295];

<TYPE> – тип атрибута OSPF Metric, принимает значения type-1 и type-2 (актуально только в режиме конфигурирования CONFIG-OSPF):

- type-1 – устанавливает тип метрики E1 для анонсируемого маршрута;
- type-2 – устанавливает тип метрики E2 для анонсируемого маршрута.

<METRIC> – значение атрибута OSPF Metric, принимает значения [0..65535] (актуально только в режиме конфигурирования CONFIG-OSPF);

<NAME> – имя маршрутной карты, которая в дальнейшем будет использоваться для фильтрации и модификации анонсируемых BGP-маршрутов, задаётся строкой до 31 символа;

<LEVEL> – уровень протокола IS-IS, в который происходит анонсирование маршрутов (актуально только в режиме конфигурирования CONFIG-ISIS):

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-BGP-FAMILY

CONFIG-BGP-VRF-FAMILY

CONFIG-OSPF

CONFIG-IPV6-OSPF

CONFIG-ISIS

CONFIG-RIP

CONFIG-RIPNG

### Пример

```
esr(config-bgp-af)# redistribute bgp 30
esr(config-ospf)# redistribute bgp 35
esr(config-rip)# redistribute bgp 300
```

**redistribute connected**

Данной командой включается анонсирование напрямую подключенных подсетей.

Использование отрицательной формы команды (no) отключает анонсирование напрямую подключенных подсетей.

**Синтаксис**

```
redistribute connected [metric <TYPE> <METRIC>][ route-map <NAME> ] [ <LEVEL> ]
no redistribute connected
```

**Параметры**

<TYPE> – тип атрибута OSPF Metric, принимает значение type-1 и type-2 (актуально только в режиме конфигурирования CONFIG-OSPF):

- type-1 – устанавливает тип метрики E1 для анонсируемого маршрута;
- type-2 – устанавливает тип метрики E2 для анонсируемого маршрута.

<METRIC> – значение атрибута OSPF Metric, принимает значения [0..65535] (актуально только в режиме конфигурирования CONFIG-OSPF).

<NAME> – имя маршрутной карты, которая в дальнейшем будет использоваться для фильтрации и модификации анонсируемых напрямую подключенных подсетей, задаётся строкой до 31 символа.

<LEVEL> – уровень протокола IS-IS, в который происходит анонсирование маршрутов (актуально только в режиме конфигурирования CONFIG-ISIS):

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-BGP-FAMILY

CONFIG-BGP-VRF-FAMILY

CONFIG-OSPF

CONFIG-IPV6-OSPF

CONFIG-ISIS

CONFIG-RIP

CONFIG-RIPNG

**Пример**

```
esr(config-rip)# redistribute connected
```

## redistribute ipv6 bgp

Данной командой включается анонсирование IPv6-маршрутов автономной системы BGP.

Использование отрицательной формы команды (no) отключает анонсирование IPv6-маршрутов автономной системы BGP.

### Синтаксис

```
redistribute ipv6 bgp <AS> [ route-map <NAME> ] [ <LEVEL> ]
no redistribute ipv6 bgp <AS>
```

### Параметры

<AS> – номер автономной системы, из которой будут анонсироваться IPv6-маршруты, может принимать значения [1..4294967295];

<NAME> – имя маршрутной карты, которая в дальнейшем будет использоваться для фильтрации и модификации анонсируемых BGP-IPv6-маршрутов, задаётся строкой до 31 символа.

<LEVEL> – уровень протокола IS-IS, в который происходит анонсирование маршрутов (актуально только в режиме конфигурирования CONFIG-ISIS):

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-ISIS

### Пример

```
esr(config-isis)# redistribute ipv6 bgp 30 level 2
```

## redistribute ipv6 ospf

Данной командой включается анонсирование маршрутов из базы OSPFv3-процесса согласно выбранным условиям.

Использование отрицательной формы команды (no) отключает анонсирование маршрутов из базы OSPFv3-процесса.

### Синтаксис

```
redistribute ipv6 ospf <ID> [ <ROUTE-TYPE1> ] [ <ROUTE-TYPE2> ] [ <ROUTE-TYPE3> ]
[ <ROUTE-TYPE4> ] [ route-map <NAME> ]
no redistribute ipv6 ospf <ID>
```

## Параметры

<ID> – номер процесса, может принимать значение [1..65535].

<ROUTE-TYPE> – типы маршрутов, которые будут анонсироваться:

- intra-area – анонсирование маршрутов OSPFv3-процесса в пределах зоны;
- inter-area – анонсирование маршрутов OSPFv3-процесса между зонами;
- external1 – анонсирование внешних маршрутов OSPFv3-формата 1;
- external2 – анонсирование внешних маршрутов OSPFv3-формата 2.

<NAME> – имя маршрутной карты, которая в дальнейшем будет использоваться для фильтрации и модификации анонсируемых OSPFv3-маршрутов, задаётся строкой до 31 символа.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ISIS

## Пример

```
esr(config-isis)# redistribute ipv6 ospf 10 external2
```

## redistribute isis

Данной командой включается анонсирование маршрутов из базы ISIS согласно выбранным условиям. Использование отрицательной формы команды (no) отключает анонсирование маршрутов из базы ISIS.

- ❗ При редистрибьюции маршрутов из базы OSPF-процесса iBGP-пиру по умолчанию BGP next-hop атрибут изменен не будет.

## Синтаксис

```
redistribute isis <ID> <ROUTE-TYPE> [metric <TYPE> <METRIC>] [ route-map <NAME> ]
no redistribute isis <ID>
```

## Параметры

<ID> – номер процесса, может принимать значение [1..65535].

<ROUTE-TYPE> – тип маршрута:

- level-1 – анонсирование маршрутов ISIS-процесса уровня 1;
- level-2 – анонсирование маршрутов ISIS-процесса уровня 2;
- inter-area – анонсирование межзоновых маршрутов IS-IS-процесса.

<TYPE> – тип атрибута OSPF Metric, принимает значение type-1 и type-2 (актуально только в режиме конфигурирования CONFIG-OSPF):

- type-1 – устанавливает тип метрики E1 для анонсируемого маршрута;
- type-2 – устанавливает тип метрики E2 для анонсируемого маршрута.



<METRIC> – значение атрибута OSPF Metric, принимает значения [0..65535] (актуально только в режиме конфигурирования CONFIG-OSPF);

<NAME> – имя маршрутной карты, которая в дальнейшем будет использоваться для фильтрации и модификации анонсируемых IS-IS-маршрутов, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-BGP-FAMILY

CONFIG-BGP-VRF-FAMILY

CONFIG-OSPF

CONFIG-IPV6-OSPF

CONFIG-ISIS

CONFIG-RIP

CONFIG-RIPNG

### Пример

```
esr(config-bgp-af)# redistribute isis 1 level-1 level-2
```

### redistribute ospf

Данной командой включается анонсирование маршрутов из базы OSPF-процесса согласно выбранным условиям.

Использование отрицательной формы команды (no) отключает анонсирование маршрутов из базы OSPF-процесса.

- i При редистрибьюции маршрутов из базы OSPF-процесса iBGP-пиру по умолчанию BGP next-hop атрибут изменен не будет.

### Синтаксис

```
redistribute ospf <ID> <ROUTE-TYPE1> [ <ROUTE-TYPE2> ] [ <ROUTE-TYPE3> ] [ <ROUTE-TYPE4> ] [metric <TYPE> <METRIC>] [ route-map <NAME> ]
```

```
no redistribute ospf <ID>
```

### Параметры

<ID> – номер процесса, может принимать значение [1..65535];

<ROUTE-TYPE> – типы маршрутов, которые будут анонсироваться:

- intra-area – анонсирование маршрутов OSPF-процесса в пределах зоны;
- inter-area – анонсирование маршрутов OSPF-процесса между зонами;
- external1 – анонсирование внешних маршрутов OSPF-формата 1;
- external2 – анонсирование внешних маршрутов OSPF-формата 2.

<TYPE> – тип атрибута OSPF Metric, принимает значение type-1 и type-2 (актуально только в режиме конфигурирования CONFIG-OSPF):

- type-1 – устанавливает тип метрики E1 для анонсируемого маршрута;
- type-2 – устанавливает тип метрики E2 для анонсируемого маршрута.

<METRIC> – значение атрибута OSPF Metric, принимает значения [0..65535] (актуально только в режиме конфигурирования CONFIG-OSPF);

<NAME> – имя маршрутной карты, которая в дальнейшем будет использоваться для фильтрации и модификации анонсируемых OSPF-маршрутов, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-BGP-FAMILY

CONFIG-BGP-VRF-FAMILY

CONFIG-OSPF

CONFIG-IPV6-OSPF

CONFIG-ISIS

CONFIG-RIP

CONFIG-RIPNG

### Пример

```
esr(config-bgp-af)# redistribute ospf 10 external2
```

### redistribute rip

Данной командой включается анонсирование маршрутов из базы маршрутов RIP.

Использование отрицательной формы команды (no) отключает анонсирование маршрутов из базы RIP.

- ❗ При редистрибуции маршрутов из базы OSPF-процесса iBGP-пиру по умолчанию BGP next-hop атрибут изменен не будет.

### Синтаксис

```
redistribute rip [metric <TYPE> <METRIC>] [ route-map <NAME> ] [ <LEVEL> ]
```

```
no redistribute rip
```

### Параметры

<TYPE> – тип атрибута OSPF Metric, принимает значение type-1 и type-2 (актуально только в режиме конфигурирования CONFIG-OSPF):

- type-1 – устанавливает тип метрики E1 для анонсируемого маршрута;
- type-2 – устанавливает тип метрики E2 для анонсируемого маршрута.

<METRIC> – значение атрибута OSPF Metric, принимает значения [0..65535] (актуально только в режиме конфигурирования CONFIG-OSPF);

<NAME> – имя маршрутной карты, которая в дальнейшем будет использоваться для фильтрации и модификации анонсируемых RIP-маршрутов, задаётся строкой до 31 символа.

<LEVEL> – уровень протокола IS-IS, в который происходит анонсирование маршрутов (актуально только в режиме конфигурирования CONFIG-ISIS):

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-BGP-FAMILY

CONFIG-BGP-VRF-FAMILY

CONFIG-OSPF

CONFIG-IPV6-OSPF

CONFIG-ISIS

### Пример

```
esr(config-bgp)# redistribute rip
```

### redistribute static

Данной командой включается анонсирование статических IPv4/IPv6-маршрутов.

Использование отрицательной формы команды (no) отключает анонсирование статических маршрутов.

### Синтаксис

```
redistribute static [metric <TYPE> <METRIC>] [ route-map <NAME> ] [ <LEVEL> ]
no redistribute static
```

### Параметры

<TYPE> – тип атрибута OSPF Metric, принимает значение type-1 и type-2 (актуально только в режиме конфигурирования CONFIG-OSPF):

- type-1 – устанавливает тип метрики E1 для анонсируемого маршрута;
- type-2 – устанавливает тип метрики E2 для анонсируемого маршрута.

<METRIC> – значение атрибута OSPF Metric, принимает значения [0..65535] (актуально только в режиме конфигурирования CONFIG-OSPF);

<NAME> – имя маршрутной карты, которая в дальнейшем будет использоваться для фильтрации и модификации анонсируемых статических маршрутов, задаётся строкой до 31 символа;

<LEVEL> – уровень протокола IS-IS, в который происходит анонсирование маршрутов (актуально только в режиме конфигурирования CONFIG-ISIS):

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне;

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-BGP-FAMILY

CONFIG-BGP-VRF-FAMILY

CONFIG-OSPF

CONFIG-IPV6-OSPF

CONFIG-ISIS

CONFIG-RIP

CONFIG-RIPNG

### Пример

```
esr(config-bgp)# redistribute static
```

## Маршрутизация на основе политик (PBR)

- action
- action add community
- action add extcommunity
- action remove community
- action remove extcommunity
- action set as-path prepend
- action set as-path replace
- action set community
- action set extcommunity
- action set ip bgp-next-hop
- action set ip next-hop
- action set ip next-hop verify-availability
- action set ipv6 bgp-next-hop
- action set ipv6 next-hop
- action set local-preference
- action set metric bgp
- action set metric isis
- action set metric ospf
- action set metric rip
- action set origin
- action set tag ospf
- action set tag rip
- action set weight bgp
- description
- ip local policy route-map
- ip policy route-map
- match as-path
- match community
- match extcommunity
- match ip access-group
- match ip address
- match ip address object-group
- match ip bgp next-hop
- match ip next-hop
- match ip route-source
- match ipv6 address
- match ipv6 address object-group
- match ipv6 bgp next-hop
- match ipv6 next-hop
- match ipv6 route-source
- match metric bgp
- match metric ospf
- match metric rip
- match tag ospf
- match tag rip
- match weight bgp
- route-map
- route-map
- rule
- show ip route-map

**action**

Данная команда используется для указания действия, которое должно быть применено для маршрутной информации, удовлетворяющей заданным критериям. Используется только для фильтрации маршрутов протоколов динамической маршрутизации и не имеет действия при конфигурировании PBR.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
action <ACT>
```

```
no action
```

**Параметры**

<ACT> – назначаемое действие:

- permit – прием/анонсирование маршрутной информации разрешено;
- deny – прием/анонсирование маршрутной информации запрещено.

**Значение по умолчанию**

permit

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-ROUTE-MAP-RULE

**Пример**

```
esr(config-route-map-rule)# action deny
```

**action add community**

Данной командой задается значение атрибута BGP Community, которое будет добавлено в существующий список.

Использование отрицательной формы команды (no) отменяет назначение.

**Синтаксис**

```
action add community { <COMMUNITY-LIST> | no-advertise | no-export }
```

```
no action add community
```

**Параметры**

<COMMUNITY-LIST> – список community, задается в виде AS:N,AS:N,AS:N, где AS-часть принимает значения [0..65535], N-часть принимает значения [0..65535]. Можно указать до 64 community;

no-advertise – при указании команды маршруты, которые передаются с данным значением атрибута community, не должны анонсироваться другим BGP-соседям;

no-export – при указании команды маршруты, которые передаются с таким значением атрибута community, не должны анонсироваться за пределы конфедерации (автономная система, которая не является частью конфедерации, считается конфедерацией). То есть, маршруты не анонсируются eBGP-соседям, но анонсируются внешним соседям в конфедерации.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-ROUTE-MAP-RULE

```
esr(config-route-map-rule)# action add community no-advertise
```

### action add extcommunity

Данной командой задается значение атрибута BGP ExtCommunity, которое будет установлено в существующий список.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
action add extcommunity <EXTCOMMUNITY-LIST>
```

```
no action add extcommunity
```

### Параметры

<EXTCOMMUNITY-LIST> – список community, задаётся в виде KIND:AS:N,KIND:AS:N,KIND:AS:N, где:

- KIND – тип extcommunity, принимает значения rt (Route Target) и ro (Route Origin);
- AS – номер автономной системы, принимает значения [1..4294967295];
- N – номер extcommunity, определяющий политику маршрутизации трафика, принимает значения [1..65535].

Можно указать до 64 ExtCommunity.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-ROUTE-MAP-RULE

```
esr(config-route-map-rule)# action add extcommunity rd:65500:100
```

### action remove community

Данной командой удаляется значение атрибута BGP Community в существующем списке.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
action remove community { <COMMUNITY-LIST> | no-advertise | no-export }
no action remove community
```

## Параметры

<COMMUNITY-LIST> – список community, задаётся в виде AS:N,AS:N,AS:N, где AS-часть принимает значения [0..65535], N-часть принимает значения [0..65535]. Можно указать до 64 community;

no-advertise – при указании команды маршруты, которые передаются с данным значением атрибута community, не должны анонсироваться другим BGP-соседям;

no-export – при указании команды маршруты, которые передаются с таким значением атрибута community, не должны анонсироваться за пределы конфедерации (автономная система, которая не является частью конфедерации, считается конфедерацией). То есть, маршруты не анонсируются eBGP-соседям, но анонсируются внешним соседям в конфедерации.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP-RULE

```
esr(config-route-map-rule)# action remove community 65500:100
```

## action remove extcommunity

Данной командой удаляется значение атрибута BGP ExtCommunity в существующем списке.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
action remove extcommunity <EXTCOMMUNITY-LIST>
no action remove extcommunity
```

## Параметры

<EXTCOMMUNITY-LIST> – список community, задаётся в виде KIND:AS:N,KIND:AS:N,KIND:AS:N, где:

- KIND – тип extcommunity, принимает значения rt (Route Target) и ro (Route Origin);
- AS – номер автономной системы, принимает значения [1..4294967295];
- N – номер extcommunity, определяющий политику маршрутизации трафика, принимает значения [1..65535].

Можно указать до 64 ExtCommunity.

## Необходимый уровень привилегий

10



## Командный режим

### CONFIG-ROUTE-MAP-RULE

```
esr(config-route-map-rule)# action remove extcommunity rt:65500:1
```

### action set as-path prepend

Данной командой устанавливается значение атрибута BGP AS-Path, которое будет добавляться в начало списка автономных систем в маршруте. При конфигурировании правила, управляемого средствами Tracking, также имеется возможность установить значение по умолчанию после ключевого слова default. Это значение будет установлено в случае если условия объекта отслеживания не будут выполнены.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
action set as-path prepend <AS-PATH> [ track <TRACK-ID> default <AS-PATH> ]
no action set as-path prepend
```

### Параметры

<AS-PATH> – список номеров автономных систем, который будет добавлен к текущему значению в маршруте. Задаётся в виде AS,AS,AS, где каждая часть принимает значения [1..4294967295]. Можно указать до 10 номеров автономных систем.

<TRACK-ID> – идентификатор объекта отслеживания, при выполнении всех условий которого будет исполняться указанное действие. Изменяется в диапазоне [1..100].

### Необходимый уровень привилегий

10

## Командный режим

### CONFIG-ROUTE-MAP-RULE

### Пример

```
esr(config-route-map-rule)# action set as-path prepend 100,200,300
```

### action set as-path replace

Данной командой включается опция замены номера или последовательности номеров AS в атрибуте BGP AS-Path маршрута на локальный номер AS.

Использование отрицательной формы команды (no) отменяет замену.

### Синтаксис

```
action set as-path replace { any | <AS-PATH> }
no action set as-path replace
```

## Параметры

any – заменять любой номер автономной системы;

<AS-PATH> – список номеров автономных систем, который будет заменён на локальный номер AS. Задаётся в виде AS,AS,AS, где каждая часть принимает значения [1..4294967295]. Можно указать до 10 номеров автономных систем.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# action set as-path prepend 100,200,300
```

## action set community

Данной командой задаётся значение атрибута BGP Community, которое будет установлено в новый список (существующий список будет перезаписан).

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
action set community { <COMMUNITY-LIST> | no-advertise | no-export }
```

```
no action set community
```

## Параметры

<COMMUNITY-LIST> – список community, задаётся в виде AS:N,AS:N,AS:N, где AS-часть принимает значения [0..65535], N часть принимает значения [0..65535]. Можно указать до 64 community;

no-advertise – при указании команды маршруты, которые передаются с данным значением атрибута community, не должны анонсироваться другим BGP-соседям;

no-export – при указании команды маршруты, которые передаются с таким значением атрибута community, не должны анонсироваться за пределы конфедерации (автономная система, которая не является частью конфедерации, считается конфедерацией). То есть, маршруты не анонсируются eBGP-соседям, но анонсируются внешним соседям в конфедерации.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# action set community no-advertise
```

### action set extcommunity

Данной командой задается значение атрибута BGP ExtCommunity, которое будет установлено в новый список (существующий список будет перезаписан).

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
action set extcommunity <EXTCOMMUNITY-LIST>
```

```
no action set extcommunity
```

### Параметры

<EXTCOMMUNITY-LIST> – список community, задаётся в виде KIND:AS:N,KIND:AS:N,KIND:AS:N, где:

- KIND – тип extcommunity, принимает значения rt (Route Target) и ro (Route Origin);
- AS – номер автономной системы, принимает значения [1..4294967295];
- N – номер extcommunity, определяющий политику маршрутизации трафика, принимает значения [1..65535].

Можно указать до 64 ExtCommunity.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# action set extcommunity ro:435:6
```

### action set ip bgp-next-hop

Данной командой задается значение атрибута BGP Next-Hop, которое будет установлено в маршруте при получении или анонсировании по BGP.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
action set ip bgp-next-hop <ADDR>
```

```
no action set ip next-hop
```

## Параметры

<ADDR> – IP-адрес шлюза, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# action set ip bgp-next-hop 10.100.100.1
```

## action set ip next-hop

Данной командой задается значение Next-Hop, которое будет установлено в маршруте, полученном по BGP.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
action set ip next-hop { <NEXTHOP> | blackhole | unreachable | prohibit }
```

```
no action set ip next-hop
```

## Параметры

<NEXTHOP> – IP-адрес шлюза задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

- blackhole – при указании команды пакеты до данной подсети будут удаляться устройством без отправки уведомлений отправителю;
- unreachable – при указании команды пакеты до данной подсети будут удаляться устройством, отправитель получит в ответ ICMP Destination unreachable (Host unreachable, code 1);
- prohibit – при указании команды пакеты до данной подсети будут удаляться устройством, отправитель получит в ответ ICMP Destination unreachable (Communication administratively prohibited, code 13).

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# action set ip next-hop prohibit
```

## action set ip next-hop verify-availability

Данной командой задается Next-Хор для пакетов, которые попадают под критерия в указанном списке доступа (ACL).

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис


```
action set ip next-hop verify-availability <NEXTHOP> <METRIC> [ track <TRACK-ID> ]
no action set ip next-hop verify-availability { <NEXTHOP> | all }
```


### Параметры

<NEXTHOP> – IP-адрес шлюза, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

[METRIC] – метрика маршрута, принимает значения [1..255];

<TRACK-ID> – идентификатор tracking объекта, при выполнении всех условий которого будет исполняться указанное действие. Изменяется в диапазоне [1..100].

 Команда *action set ip next-hop* с ключом *verify-availability* применяется для создания route-map, которые будут использоваться для Policy-Based-Routing, а не для изменения next-hop в маршрутах, полученных по протоколу динамической маршрутизации.

 Ключ *verify-availability* указывает, что данный *action* будет работать, только если активен IP-интерфейс маршрутизатора терминирующий подсеть, к которой относится указанный в команде <NEXTHOP> .

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-ROUTE-MAP-RULE

### Пример

```
esr(config-route-map-rule)# action set ip next-hop verify-availability 1.1.1.1 25
```

## action set ipv6 bgp-next-hop

Данной командой задается значение атрибута BGP Next-Хор для IPv6, которое будет установлено в маршруте при анонсировании по BGP.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
action set ipv6 bgp-next-hop <ADDR>
no action set ipv6 next-hop
```

## Параметры

<IPv6-ADDR> – IPv6-адрес шлюза, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# action set ipv6 bgp-next-hop 2002::765:1
```

## action set ipv6 next-hop

Данной командой задается значение Next-Hop для IPv6, которое будет установлено в маршруте, полученном по BGP.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
action set ipv6 next-hop <NEXTHOP>
```

```
no action set ipv6 next-hop
```

## Параметры

<NEXTHOP> – IPv6-адрес шлюза, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# action set ipv6 next-hop 55::205:2
```

## action set local-preference

Данной командой задается значение атрибута BGP Local Preference, которое будет установлено в маршруте, а при использовании параметра increment/decrement – дельта, на которую будет увеличен или уменьшен этот атрибут относительно исходного значения соответственно.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
action set local-preference {<PREFERENCE> | increment < VALUE > | decrement < VALUE >}
no action set local-preference
```

### Параметры

<VALUE> – значение дельты изменения атрибута BGP Local Preference относительно исходного значения. Принимает значение [1..2147483647]. Если в результате применения операции increment/decrement значение метрики выйдет за допустимый диапазон, значение Local Preference принимается равным максимально или минимально допустимому значению соответственно.

<PREFERENCE> – значение атрибута BGP Local Preference, принимает значения [1..2147483647].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-ROUTE-MAP-RULE

### Пример

```
esr(config-route-map-rule)# action set local-preference 120
```

### action set metric bgp

Данной командой задается значение атрибута BGP MED, которое будет установлено в маршруте, а при использовании параметра increment/decrement – дельта, на которую будет увеличен или уменьшен этот атрибут относительно исходного значения соответственно. При конфигурировании правила, управляемого средствами Tracking, также имеется возможность установить значение по умолчанию после ключевого слова default. Это значение будет установлено в случае, если условия объекта отслеживания не будут выполнены.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
action set metric bgp {< METRIC > | increment < VALUE > | decrement < VALUE >} [ track
<TRACK-ID> [default <METRIC>] ]
no action set metric bgp
```

### Параметры

<VALUE> – значение дельты изменения атрибута BGP MED относительно исходного значения. Принимает значение [0..4294967295]. Если в результате применения операции increment/decrement значение метрики выйдет за допустимый диапазон, значение BGP MED принимается равным максимально или минимально допустимому значению соответственно.

<METRIC> – значение атрибута BGP MED, принимает значения [0..4294967295];

<TRACK-ID> – идентификатор объекта отслеживания, при выполнении всех условий которого будет исполняться указанное действие. Изменяется в диапазоне [1..100].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-ROUTE-MAP-RULE

### Пример

```
esr(config-route-map-rule)# action set metric bgp 10
```

### action set metric isis

Данной командой задается значение атрибута IS-IS Metric, которое будет установлено в маршруте. Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
action set metric isis <METRIC>  
no action set metric isis
```

### Параметры

<METRIC> – значение атрибута IS-IS Metric, принимает значения [1..16777215].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-ROUTE-MAP-RULE

### Пример

```
esr(config-route-map-rule)# action set metric isis 10
```

### action set metric ospf

Данной командой задается значение атрибута OSPF Metric, которое будет установлено в маршруте. Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
action set metric ospf <TYPE> <METRIC>  
no action set metric ospf
```



## Параметры

<TYPE> – тип атрибута OSPF Metric, принимает значение type-1 и type-2.

<METRIC> – значение атрибута OSPF Metric, принимает значения [0..65535].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# action set metric ospf type-1 10
```

## action set metric rip

Данной командой задается значение атрибута RIP Metric, которое будет установлено в маршруте.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
action set metric rip <METRIC>
```

```
no action set metric rip
```

## Параметры

<METRIC> – значение атрибута RIP Metric, принимает значения [0..16].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# action set metric rip 5
```

## action set origin

Данной командой задается значение атрибута BGP Origin, которое будет установлено в маршруте.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
action set origin <ORIGIN>
no action set origin
```

### Параметры

<ORIGIN> – значение атрибута BGP Origin, принимает следующие значения:

- egr – маршрут выучен по протоколу Exterior Gateway Protocol (EGP);
- igp – маршрут получен внутри исходной автономной системы;
- incomplete – маршрут выучен другим образом.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-ROUTE-MAP-RULE

### Пример

```
esr(config-route-map-rule)# action set origin igp
```

### action set tag ospf

Данной командой задается значение атрибута OSPF Tag, которое будет установлено в маршруте. Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
action set tag ospf <TAG>
no action set tag ospf
```

### Параметры

<TAG> – значение атрибута OSPF Tag, принимает значения [0..4294967295].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-ROUTE-MAP-RULE

### Пример

```
esr(config-route-map-rule)# action set tag ospf 20
```

### action set tag rip

Данной командой задается значение атрибута RIP Tag, которое будет установлено в маршруте. Использование отрицательной формы команды (no) отменяет назначение.

#### Синтаксис

```
action set tag rip <RIP>  
no action set tag rip
```

#### Параметры

<RIP> – значение атрибута RIP Tag, принимает значения [0..65535].

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-ROUTE-MAP-RULE

#### Пример

```
esr(config-route-map-rule)# action set tag rip 20
```

### action set weight bgp

Данной командой задается значение атрибута BGP weight, которое будет установлено в маршруте. Использование отрицательной формы команды (no) отменяет назначение.

#### Синтаксис

```
action set weight bgp <WEIGHT>  
no action set weight bgp
```

#### Параметры

<WEIGHT> – значение атрибута BGP weight, принимает значения [0..65535].

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# action set weight bgp 200
```

## description

Данная команда используется для изменения описания конфигурируемого правила маршрутной карты. Использование отрицательной формы команды (no) удаляет установленное описание.

## Синтаксис

```
description <DESCRIPTION>
no description
```

## Параметры

<DESCRIPTION> – описание правила маршрутной карты, задаётся строкой до 255 символов.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# description "Drop Local NETs"
```

## ip local policy route-map

Данной командой назначается политика маршрутизации на основе списков доступа (ACL). Данная политика маршрутизации работает для пакетов, которые генерирует сам маршрутизатор.

Использование отрицательной формы команды (no) удаляет политику маршрутизации.

## Синтаксис

```
ip local policy [vrf <VRF>] route-map <NAME>
no local ip policy [vrf <VRF>] route-map
```

## Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра политика маршрутизации будет работать только в определенном VRF;

<NAME> – имя сконфигурированной политики маршрутизации, задаётся строкой до 31 символа.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip local policy route-map drop-local-net
```

**ip policy route-map**

Данной командой на интерфейс назначается политика маршрутизации на основе списков доступа (ACL). Данная политика маршрутизации работает для пакетов, проходящих через интерфейс, на котором она назначена.

Использование отрицательной формы команды (no) удаляет политику маршрутизации.

**Синтаксис**

```
ip policy route-map <NAME>
```

```
no ip policy route-map
```

**Параметры**

<NAME> – имя сконфигурированной политики маршрутизации, задаётся строкой до 31 символа.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-CELLULAR-MODEM

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-VTI

## Пример

```
esr(config-if-gi)# ip policy route-map drop-local-net in
```

## match as-path

Данной командой устанавливается значение атрибута BGP AS-Path в маршруте, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
match as-path { [ begin | contain | end ] <AS-PATH> | empty | regex <REGEX> }
```

```
no match as-path
```

## Параметры

<AS-PATH> – список номеров автономных систем, задаётся в виде AS,AS,AS, где каждая часть принимает значения [1..4294967295]. Можно указать до 10 номеров автономных систем.

Оptionальные параметры, с помощью которых задаётся частичное соответствие атрибута:

- begin – значение атрибута начинается с указанного списка номеров автономных систем;
- end – значение атрибута оканчивается на указанный список номеров автономных систем;
- contain – значение атрибута содержит указанный список номеров автономных систем;
- empty – значение атрибута пусто;
- regex – значение атрибута соответствует регулярному выражению, где <REGEX> – регулярное выражение, задаётся по стандарту POSIX-Extended Regular Expressions.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# match as-path begin 100,200,300
```

## match community

Данной командой устанавливается значение атрибута BGP Community в маршруте, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
match community { <COMMUNITY-LIST> | regex <REGEX> }
```

```
no match community
```

## Параметры

<COMMUNITY-LIST> – список community, задаётся в виде AS:N,AS:,AS, где AS часть принимает значения [0..65535], N часть принимает значения [0..65535]. Можно указать до 64 community.

Оptionальные параметры, с помощью которых задаётся частичное соответствие атрибута:

- regex – значение атрибута соответствует регулярному выражению, где <REGEX> – регулярное выражение, задаётся по стандарту POSIX-Extended Regular Expressions.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# match community 100:1,200:3,300:65000
```

## match extcommunity

Данной командой устанавливается значение атрибута BGP ExtCommunity в маршруте, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
match extcommunity { <EXTCOMMUNITY-LIST> | regex <REGEX> }
no match extcommunity
```

## Параметры

<EXTCOMMUNITY-LIST> – список extcommunity, задаётся в виде KIND:AS:N, KIND:AS:N, KIND:AS:N, где:

- KIND – тип extcommunity, принимает значения rt (Route Target) или ro (Route Origin);
- AS – номер автономной системы, принимает значения [1..4294967295];
- N – номер extcommunity, определяющий политику маршрутизации трафика, принимает значения [1..65535].

Можно указать до 64 extcommunity.

Оptionальные параметры, с помощью которых задаётся частичное соответствие атрибута:

- regex – значение атрибута соответствует регулярному выражению, где <REGEX> – регулярное выражение, задаётся по стандарту POSIX-Extended Regular Expressions.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# match extcommunity ro:435:6
```

### match ip access-group

Данной командой устанавливается ACL-группа, для которой должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
match ip access-group <NAME>
```

```
no match ip access-group
```

### Параметры

<NAME> – имя списка контроля доступа, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# match ip access-group ACCESS
```

### match ip address

Данной командой устанавливается адреса маршрутов, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
match ip address <ADDR/LEN> [ { eq <LEN> | le <LEN> | ge <LEN> [ le <LEN> ] } ]
```

```
no match ip address prefix
```

### Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<LEN> – длина префикса, принимает значения [1..32];

eq – при указании команды длина префикса должна соответствовать указанной;

le – при указании команды длина префикса должна быть меньше либо соответствовать указанной;

ge – при указании команды длина префикса должна быть больше либо соответствовать указанной.



**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-ROUTE-MAP-RULE

**Пример**

```
esr(config-route-map-rule)# match ip address 192.168.0.0/32 ge 16
```

**match ip address object-group**

Данной командой устанавливается профиль IP-адресов, содержащий значения подсетей назначения в маршруте, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

**Синтаксис**

```
match ip address object-group <OBJ-GROUP- NETWORK -NAME> [ { eq <LEN> | le <LEN> | ge <LEN> [ le <LEN> ]
```

```
no match ip address object-group
```

**Параметры**

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, который должен содержать префиксы подсетей назначения, задаётся строкой до 31 символа;

eq – при указании команды длина префикса должна соответствовать указанной;

le – при указании команды длина префикса должна быть меньше либо соответствовать указанной;

ge – при указании команды длина префикса должна быть больше либо соответствовать указанной.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-ROUTE-MAP-RULE

**Пример**

```
esr(config-route-map-rule)# match ip address object-group local_nets
```

Данной командой устанавливается профиль IP-адресов, содержащий значения Next-Hop в таблице маршрутизации (show ip route) маршрутизатора, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

**Синтаксис**

```
match ip next-hop object-group <OBJ-GROUP-NETWORK-NAME>
no match ip next-hop
```

**Параметры**

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, который должен содержать диапазоны IP-адресов шлюзов, задаётся строкой до 31 символа.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-ROUTE-MAP-RULE

**Пример**

```
esr(config-route-map-rule)# match ip next-hop object-group block_nexthop
```

**match ip bgp next-hop**

Данной командой устанавливается профиль IP-адресов, содержащий значения BGP атрибута Next-Hop, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

**Синтаксис**

```
match ip bgp next-hop object-group <OBJ-GROUP-NETWORK-NAME>
no match ip bgp next-hop
```

**Параметры**

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, который должен содержать диапазоны IP-адресов шлюзов, задаётся строкой до 31 символа.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-ROUTE-MAP-RULE

**Пример**

```
esr(config-route-map-rule)# match ip bgp next-hop object-group Next-hop
```

## match ip next-hop

Данной командой устанавливается профиль IP-адресов, содержащий значения next-hop в таблице маршрутизации (show ip route) маршрутизатора, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
match ip next-hop object-group <OBJ-GROUP-NETWORK-NAME>
```

```
no match ip next-hop
```

### Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, который должен содержать диапазоны IP-адресов шлюзов, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-ROUTE-MAP-RULE

### Пример

```
esr(config-route-map-rule)# match ip next-hop object-group block_nexthop
```

## match ip route-source

Командой устанавливается профиль IP-адресов. Профиль содержит IP-адреса маршрутизатора, анонсировавшего маршрут. Используется для фильтрации по IP-адресу источника при анонсировании маршрутной информации.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
match ip route-source object-group <OBJ-GROUP-NETWORK-NAME>
```

```
no match ip route-source
```

### Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, который должен содержать диапазоны IP-адресов источника маршрутной информации, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

10

## Командный режим

### CONFIG-ROUTE-MAP-RULE

#### Пример

```
esr(config-route-map-rule)# match ip route-source object-group source_routers
```

#### match ipv6 address

Данной командой устанавливается адреса IPv6-маршрутов, для которых должно срабатывать правило. Использование отрицательной формы команды (no) отменяет назначение.

#### Синтаксис

```
match ipv6 address <IPV6-ADDR/LEN> [ { eq <LEN> | le <LEN> | ge <LEN> [ le <LEN> ] } ]
no match ipv6 address prefix
```

#### Параметры

<IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<LEN> – длина префикса, принимает значения [1..128];

eq – при указании команды длина префикса должна соответствовать указанной;

le – при указании команды длина префикса должна быть меньше либо соответствовать указанной;

ge – при указании команды длина префикса должна быть больше либо соответствовать указанной.

#### Необходимый уровень привилегий

10

## Командный режим

### CONFIG-ROUTE-MAP-RULE

#### Пример

```
esr(config-route-map-rule)# match ipv6 address FF02:0:0:0:0:1:FF00::/104
```

#### match ipv6 address object-group

Данной командой устанавливается профиль IPv6-адресов, содержащий значения подсетей назначения в маршруте, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

#### Синтаксис

```
match ipv6 address object-group <OBJ-GROUP- NETWORK -NAME> [ { eq <LEN> | le <LEN> | ge <LEN> [ le <LEN> ] }
```

```
no match ipv6 address object-group
```

### Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IPv6-адресов, который должен содержать префиксы подсетей назначения, задаётся строкой до 31 символа;

eq – при указании команды длина префикса должна соответствовать указанной;

le – при указании команды длина префикса должна быть меньше либо соответствовать указанной;

ge – при указании команды длина префикса должна быть больше либо соответствовать указанной.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-ROUTE-MAP-RULE

### Пример

```
esr(config-route-map-rule)# match ipv6 address object-group local_nets
```

### match ipv6 bgp next-hop

Данной командой устанавливается профиль IPv6-адресов, содержащий значения BGP атрибута Next-Hop, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
match ipv6 bgp next-hop object-group <OBJ-GROUP-NETWORK-NAME>
```

```
no match ipv6 next-hop
```

### Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IPv6-адресов, который должен содержать диапазоны IPv6-адресов шлюзов, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-ROUTE-MAP-RULE

### Пример

```
esr(config-route-map-rule)# match ipv6 bgp next-hop object-group block_next-hop
```

## match ipv6 next-hop

Данной командой устанавливается профиль IPv6-адресов, содержащий значения Next-Hop в таблице маршрутизации (show ip route) маршрутизатора, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
match ipv6 next-hop object-group <OBJ-GROUP-NETWORK-NAME>
```

```
no match ipv6 next-hop
```

### Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IPv6-адресов, который должен содержать диапазоны IPv6-адресов шлюзов, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-ROUTE-MAP-RULE

### Пример

```
esr(config-route-map-rule)# match ipv6 next-hop object-group block_nexthop
```

## match ipv6 route-source

Данной командой устанавливается профиль IPv6-адресов. Профиль содержит IPv6-адреса маршрутизатора, анонсировавшего маршрут. Используется для фильтрации по IPv6-адресу источника при анонсировании маршрутной информации.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
match ipv6 route-source object-group <OBJ-GROUP-NETWORK-NAME>
```

```
no match ipv6 route-source
```

### Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IPv6-адресов, который должен содержать диапазоны IPv6-адресов источника маршрутной информации, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP-RULE

### Пример

```
esr(config-route-map-rule)# match ipv6 route-source object-group source_routers
```

### match metric bgp

Данной командой устанавливается значение атрибута BGP MED в маршруте, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
match metric bgp <METRIC>
```

```
no match metric bgp
```

### Параметры

<METRIC> – значение атрибута BGP MED, принимает значения [0..4294967295].

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP-RULE

### Пример

```
esr(config-route-map-rule)# match metric bgp 10
```

### match metric ospf

Данной командой устанавливается значение атрибута OSPF Metric в маршруте, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
match metric ospf <TYPE> <METRIC>
```

```
no match metric ospf
```

### Параметры

<TYPE> – тип атрибута OSPF Metric, принимает значение type-1 и type-2.

<METRIC> – значение атрибута OSPF Metric, принимает значения [0..65535].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-ROUTE-MAP-RULE

### Пример

```
esr(config-route-map-rule)# match metric ospf type-1 10
```

### match metric rip

Данной командой устанавливается значение атрибута RIP Metric в маршруте, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
match metric rip <METRIC>
```

```
no match metric rip
```

### Параметры

<METRIC> – значение атрибута RIP Metric, принимает значения [0..16].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-ROUTE-MAP-RULE

### Пример

```
esr(config-route-map-rule)# match metric rip 5
```

### match tag ospf

Данной командой устанавливается значение атрибута OSPF Tag в маршруте, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
match tag ospf <TAG>
```



```
no match tag ospf
```

### Параметры

<TAG> – значение атрибута OSPF Tag, принимает значения [0..4294967295].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-ROUTE-MAP-RULE

### Пример

```
esr(config-route-map-rule)# match tag ospf 20
```

### match tag rip

Данной командой устанавливается значение атрибута RIP Tag в маршруте, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
match set tag rip <RIP>
```

```
no match set tag rip
```

### Параметры

<RIP> – значение атрибута RIP Tag, принимает значения [0..65535].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-ROUTE-MAP-RULE

### Пример

```
esr(config-route-map-rule)# match tag rip 20
```

### match weight bgp

Данной командой устанавливается значение атрибута BGP weight в маршруте, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

**Синтаксис**

```
match weight bgp <WEIGHT>
no match weight bgp
```

**Параметры**

<WEIGHT> – значение атрибута BGP weight, принимает значения [0..65535].

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-ROUTE-MAP-RULE

**Пример**

```
esr(config-route-map-rule)# match bgp weight 20
```

**route-map**

Командой добавляется фильтрация и модификация маршрутов во входящих или исходящих направлениях.

Использование отрицательной формы команды (no) отключает фильтрацию и модификацию маршрутов в соответствующем направлении.

**Синтаксис**

```
route-map <NAME> <DIRECTION>
no route-map <DIRECTION>
```

**Параметры**

<NAME> – имя сконфигурированной маршрутной карты, задаётся строкой до 31 символа;

<DIRECTION> – направление:

- in – фильтрация и модификация получаемых маршрутов;
- out – фильтрация и модификация анонсируемых маршрутов.

**Необходимый уровень привилегий**

10

**Командный режим**

```
CONFIG-BGP-GROUP
CONFIG-BGP-VRF-GROUP
CONFIG-BGP-NEIGHBOR-FAMILY
CONFIG-BGP-VRF-NEIGHBOR-FAMILY
```

CONFIG-OSPF  
CONFIG-IPV6-OSPF  
CONFIG-ISIS  
CONFIG-RIP  
CONFIG-RIPND

### Пример

```
esr(config-bgp-neighbor)# route-map drop-local-net in
```

### route-map

Данной командой создается маршрутная карта, которая в дальнейшем будет использоваться для фильтрации и модификации анонсируемых и получаемых IP-маршрутов, и осуществляется переход в режим настройки параметров маршрутной карты.

Использование отрицательной формы команды (no) удаляет маршрутную карту.

### Синтаксис

```
[no] route-map <NAME>
```

### Параметры

<NAME> – имя конфигурируемой маршрутной карты, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG


### Пример

```
esr(config)# route-map drop-local-net  
esr(config-route-map)#
```

### rule

Данной командой создается правило маршрутной карты с определённым номером и осуществляется переход в режим настройки параметров правила. Правила обрабатываются устройством в порядке возрастания номеров правил.

Использование отрицательной формы команды (no) удаляет правило по номеру либо все правила.

 Критерии отбора, описанные в правиле, подчиняются логическому "И".  
Например:

```

ESR(config)# route-map TEST
ESR(config-route-map)# rule 1
ESR(config-route-map-rule)# match community 65:100
ESR(config-route-map-rule)# match weight bgp 200
ESR(config-route-map-rule)# exit

```

Под правило попадут BGP-маршруты, в атрибутах которых community и MED находятся значения 65:100 и 200 соответственно.

Правила обрабатываются устройством в порядке возрастания номеров правил. Если маршрут подпадает под все условия (match), описанные в правиле, выполняется настроенное в правиле действие (action) и дальнейший просмотр правил не производится.

## Синтаксис

```
[no] rule <ORDER>
```

## Параметры

<ORDER> – номер правила, принимает значения [1..10000].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP

## Пример

```

esr(config-route-map)# rule 2
esr(config-route-map-rule)#

```

## show ip route-map

Данная команда используется для просмотра маршрутных карт.

## Синтаксис

```
show ip route-map <NAME> [ <ORDER> ]
```

## Параметры

<NAME> – имя маршрутной карты, задаётся строкой до 31 символа;

<ORDER> – номер правила, принимает значения [1..10000]. При указании номера правила будет показана информация только по данному правилу.

## Необходимый уровень привилегий

1

**Командный режим**

ROOT

**Пример**

```

esr# show ip route-map drop-local-net
Order:                               2
Description:                          Drop route to private nets
Matching pattern:
  Access group                        --
  AS path                             --
  Community                           --
  Extcommunity                        --
  BGP metric (MED):                  --
  Address (object-group):             local_net
  Next hop (object-group):            --
  Route source (object-group):        --
  RIP metric                          --
  RIP tag                             --
  OSPF metric type                    --
  OSPF metric                         --
  OSPF tag                            --
Actions:
  Decision:                           Deny
  Route next hop address:              --
  Route IPv6 next hop address:         --
  Route next hop:                     --
  AS path (prepend):                  --
  Community:                           --
  Extcommunity:                       --
  Local preference:                   --
  BGP next hop address:               --
  BGP IPv6 next hop address:          --
  BGP metric (MED):                   --
  Origin:                             --
  RIP metric                          --
  RIP tag                             --
  OSPF metric type                    --
  OSPF metric                         --
  OSPF tag                            --

```

## Настройка связок ключей

- [accept-lifetime](#)
- [key](#)
- [key-chain](#)
- [key-string](#)
- [send-lifetime](#)

### accept-lifetime

Данная команда определяет период времени, в течение которого данный ключ может использоваться для аутентификации принятых пакетов.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
accept-lifetime <TIME_B> <DAY_B> <MONTH_B> <YEAR_B> <TIME_E> <DAY_E> <MONTH_E> <YEAR_E>
no accept-lifetime
```

### Параметры

<TIME\_B> – устанавливаемое время начала использования ключа, задаётся в виде HH:MM:SS, где:

- HH – часы, принимает значение [0..23];
- MM – минуты, принимает значение [0..59];
- SS – секунды, принимает значение [0..59].

<DAY\_B> – день месяца начала использования ключа, принимает значения [1..31];

<MONTH\_B> – месяц начала использования ключа, принимает значения:

- January;
- February;
- March;
- April;
- May;
- June;
- July;
- August;
- September;
- October;
- November;
- December.

<YEAR\_B> – год начала использования ключа, принимает значения [2001..2037].

<TIME\_E> – устанавливаемое время окончания использования ключа, задаётся в виде HH:MM:SS, где:

- HH – часы, принимает значение [0..23];
- MM – минуты, принимает значение [0..59];
- SS – секунды, принимает значение [0..59].

<DAY\_E> – день месяца окончания использования ключа, принимает значения [1..31];

<MONTH\_E> – месяц окончания использования ключа, принимает значения:

- January;
- February;
- March;
- April;

- May;
- June;
- July;
- August;
- September;
- October;
- November;
- December.

<YEAR\_E> – год окончания использования ключа, принимает значения [2001..2037].

### Значение по умолчанию

Ключ действителен постоянно.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-KEYCHAIN-KEY

### Пример

```
esr(config-keychain-key)# accept-lifetime 16:35:00 10 May 2015 16:35:00 10 June 2021
```

### key

Данной командой добавляется ключ в связку ключей и осуществляется переход в режим настройки параметров ключа.

Использование отрицательной формы команды (no) удаляет указанный ключ.

### Синтаксис

[no] key <ID>

### Параметры

<ID> – идентификатор ключа, задается в диапазоне [0..255].

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-KEYCHAIN

### Пример

```
esr(config-keychain)# key 25
```

**key-chain**

данной командой добавляется связка ключей в систему и осуществляется переход в режим настройки параметров связки ключей.

Использование отрицательной формы команды (no) удаляет указанный список.

**Синтаксис**

```
[no] key-chain <KEYCHAIN>
```

**Параметры**

<KEYCHAIN> – идентификатор списка ключей, строка до 16 ASCII-символов.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# key-chain lock
```

**key-string**

Данной командой устанавливается пароль для аутентификации.

Использование отрицательной формы команды (no) удаляет пароль.

**Синтаксис**

```
key-string ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no key-string
```

**Параметры**

<CLEAR-TEXT> – пароль, задаётся строкой от 8 до 16 символов;

<ENCRYPTED-TEXT> – зашифрованный пароль размером от 8 байт до 16 байт (от 16 до 32 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-KEYCHAIN-KEY



## Пример

```
esr(config-keychain-key)# key-string ascii-text 123456789
esr(config-keychain-key)# key-string ascii-text encrypted CDE65039E5591FA3F1
```

## send-lifetime

Данная команда определяет период времени, в течение которого данный ключ может использоваться для аутентификации при отправке пакетов.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
send-lifetime <TIME_B> <DAY_B> <MONTH_B> <YEAR_B> <TIME_E> <DAY_E> <MONTH_E> <YEAR_E>
no send-lifetime
```

## Параметры

<TIME\_B> – устанавливаемое время начала использования ключа, задаётся в виде HH:MM:SS, где:

- HH – часы, принимает значение [0..23];
- MM – минуты, принимает значение [0..59];
- SS – секунды, принимает значение [0..59].

<DAY\_B> – день месяца начала использования ключа, принимает значения [1..31];

<MONTH\_B> – месяц начала использования ключа, принимает значения [January/February/March/April/May/June/July/August/September/October/November/December];

<YEAR\_B> – год начала использования ключа, принимает значения [2001..2037];

<TIME\_E> – устанавливаемое время окончания использования ключа, задаётся в виде HH:MM:SS, где:

- HH – часы, принимает значение [0..23];
- MM – минуты, принимает значение [0..59];
- SS – секунды, принимает значение [0..59].

<DAY\_E> – день месяца окончания использования ключа, принимает значения [1..31];

<MONTH\_E> – месяц окончания использования ключа, принимает значения [January/February/March/April/May/June/July/August/September/October/November/December];

<YEAR\_E> – год окончания использования ключа, принимает значения [2001..2037].

## Значение по умолчанию

Ключ действителен постоянно.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-KEYCHAIN-KEY

## Пример

```
esr(config-keychain-key)# send-lifetime 16:35:00 15 May 2014 16:35:00 21 June 2018
```

## Настройка параметров протокола BFD

- [clear bfd](#)
- [ip bfd idle-tx-interval](#)
- [ip bfd log-adjacency-changes](#)
- [ip bfd min-rx-interval](#)
- [ip bfd min-tx-interval](#)
- [ip bfd multiplier](#)
- [ipv6 bfd passive](#)
- [ipv6 bfd log-adjacency-changes](#)
- [ipv6 ospf bfd-enable](#)
- [show ip bfd](#)
- [show bfd neighbors](#)

### clear bfd

Данная команда перезапускает процесс BFD. В результате выполнения команды все установленные BFD-сессии (IPv4/IPv6 в GRT/VRF) будут сброшены.

### Синтаксис

```
clear bfd
```

### Параметры

Не содержит параметров.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

## Пример

```
esr# clear bfd
2022-01-04T21:00:19+00:00 %BFD-W-NEIG: Session to 192.168.39.230 Down
```

### ip bfd idle-tx-interval

Данной командой задаётся интервал, по истечении которого происходит отправка BFD-сообщения соседу. Данный параметр служит для уменьшения числа генерируемых сообщений для тех случаев, когда BFD-сосед недоступен или на нем выключен протокол BFD.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
ip bfd idle-tx-interval <TIMEOUT>
no ip bfd idle-tx-interval
```

## Параметры

<TIMEOUT> – интервал, по истечении которого происходит отправка BFD-пакета, принимает значение в миллисекундах в диапазоне:

- [200..65535] для ESR-1000/1200/1500/1511/1700/3100/3200/3200L/3300;
- [300..65535] для ESR-10/12V/12VF/15/15R/15VF/20/21/30/31/100/200.

## Значение по умолчанию

1 секунда

## Необходимый уровень привилегий

10

## Командный режим

```
CONFIG
CONFIG-IF-GI
CONFIG-IF-TE
CONFIG-IF-TWE
CONFIG-IF-FO
CONFIG-IF-HU
CONFIG-IF-SUB
CONFIG-IF-QINQ
CONFIG-IF-PORT-CHANNEL
CONFIG-BRIDGE
CONFIG-GRE
CONFIG-IP4IP4
```

## Пример

```
esr(config)# ip bfd idle-tx-interval 4000
```

## ip bfd log-adjacency-changes

Данной командой включается логирование изменений состояния BFD-протокола.

Использование отрицательной формы команды (no) отключает логирование изменений состояния BFD-протокола.

**Синтаксис**

```
ip bfd log-adjacency-changes
no ip bfd log-adjacency-changes
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip bfd log-adjacency-changes
```

**ip bfd min-rx-interval**

Данной командой задаётся минимальный интервал, по истечении которого сосед должен сгенерировать BFD-сообщение. Данный параметр анонсируется соседу.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
ip bfd min-rx-interval <TIMEOUT>
no ip bfd min-rx-interval
```

**Параметры**

<TIMEOUT> – интервал, по истечении которого должна происходить отправка BFD-сообщения соседом, принимает значение в миллисекундах в диапазоне:

- [200..65535] для ESR-1000/1200/1500/1511/1700/3100/3200/3200L/3300;
- [300..65535] для ESR-10/12V/12VF/15/15R/15VF/20/21/30/31/100/200.

**Значение по умолчанию**

- 200 миллисекунд на ESR-1000/1200/1500/1511/1700/3100/3200/3200L/3300;
- 300 миллисекунд на ESR-10/12V/12VF/15/15R/15VF/20/21/30/31/100/200.

**Необходимый уровень привилегий**

10

## Командный режим

CONFIG

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP4

## Пример

```
esr(config)# ip bfd min-rx-interval 1000
```

## ip bfd min-tx-interval

Данной командой задаётся минимальный интервал, по истечении которого происходит отправка BFD-сообщения соседу. Данный параметр используется только когда BFD-сессия активна, в иных случаях используется [ip bfd idle-tx-interval](#).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
ip bfd min-tx-interval <TIMEOUT>
```

```
no ip bfd min-tx-interval
```

## Параметры

<TIMEOUT> – интервал, по истечении которого должна происходить отправка BFD-сообщения соседу, принимает значение в миллисекундах в диапазоне:

- [200..65535] для ESR-1000/1200/1500/1511/1700/3100/3200/3200L/3300;
- [300..65535] для ESR-10/12V/12VF/15/15R/15VF/20/21/30/31/100/200.

## Значение по умолчанию

- 200 миллисекунд на ESR-1000/1200/1500/1511/1700/3100/3200/3200L/3300;
- 300 миллисекунд на ESR-10/12V/12VF/15/15R/20/21/30/31/100/200.

## Необходимый уровень привилегий

10

**Командный режим**

CONFIG  
 CONFIG-IF-GI  
 CONFIG-IF-TE  
 CONFIG-IF-TWE  
 CONFIG-IF-FO  
 CONFIG-IF-HU  
 CONFIG-IF-SUB  
 CONFIG-IF-QINQ  
 CONFIG-IF-PORT-CHANNEL  
 CONFIG-BRIDGE  
 CONFIG-GRE  
 CONFIG-IP4IP4

**Пример**

```
esr(config)# ip bfd min-tx-interval 1000
```

**ip bfd multiplier**

Данной командой задаётся число пропущенных пакетов, после достижения которого BFD-сосед считается недоступным. Время обнаружения недоступности в каждом из направлений рассчитывается из данного числа умноженного на tx/rx-интервал.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
ip bfd multiplier <COUNT>
no ip multiplier
```

**Параметры**

<COUNT> – число пропущенных пакетов, после достижения которого сосед считается недоступным, принимает значение в диапазоне [1..100].

**Значение по умолчанию**

5

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-GRE  
CONFIG-IP4IP4

### Пример

```
esr(config)# ip bfd multiplier 10
```

### ipv6 bfd passive

Данной командой BFD-сессия переводится в пассивный режим, то есть BFD-сообщения не будут отправляться до тех пор, пока не будут получены сообщения от BFD-соседа.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
[no] ip bfd passive
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Активный режим.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB

CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-GRE  
CONFIG-IP4IP4

### Пример

```
esr(config-if-gi)# ip bfd passive
```

### ipv6 bfd log-adjacency-changes

Данной командой включается логирование изменений состояния IPv6 BFD-протокола.

Использование отрицательной формы команды (no) отключает логирование изменений состояния IPv6 BFD-протокола.

### Синтаксис

```
ipv6 bfd log-adjacency-changes  
no ipv6 bfd log-adjacency-changes
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Логирование выключено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# ipv6 bfd log-adjacency-changes
```

### ipv6 ospf bfd-enable

Данной командой на интерфейсе включается протокол BFD для протокола OSPFv3.

Использование отрицательной формы команды (no), на интерфейсе отключается протокол BFD для протокола OSPFv3.



## Синтаксис

```
[no] ipv6 ospf bfd-enable
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Протокол BFD для протокола OSPF отключен.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

## Пример

```
esr(config-if-gi)# ipv6 ospf bfd-enable
```

## show ip bfd

Данная команда отображает информацию о параметрах BFD-протокола или определенных интерфейсов при использовании фильтра.

## Синтаксис

```
show ip bfd [ { interface <IF> | tunnel <TUN> } ]
```

## Параметры

<IF> – интерфейс или группы интерфейсов, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

<TUN> – имя туннеля, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#).

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
esr# show ip bfd
Minimum RX interval: 200 ms
Minimum TX interval: 200 ms
Idle TX interval: 1000 ms
Multiplier: 5 packets
Passive: No
esr# show ip bfd interface gi1/0/1
Minimum RX interval: 200 ms
Minimum TX interval: 200 ms
Idle TX interval: 1000 ms
Multiplier: 10 packets
Passive: Yes
```

### show bfd neighbors

Данная команда отображает список всех сессий, установленных (настроенных) по протоколу BFD.

### Синтаксис

```
show bfd neighbors [ vrf <NAME_VRF> ] [ {<IPv4-ADDRESS> | <IPv6-ADDRESS> | client { bfd
| bgp | static | ospf } } ]
```

### Параметры

<NAME-VRF> – имя экземпляра VRF, задается строкой до 31 символа;

<IPv4-ADDRESS> – IP-адрес соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPv6-ADDRESS> – IPv6-адрес соседа, задается в виде X:X:X:X::X, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF];

bfd – выводит информацию о настроенных (с помощью команды ip/ipv6 bfd neighbors) BFD-сессиях;

static – при указании команды отображаются BFD-сессии созданные для статических маршрутов;

ospf – выводит информацию о BFD-сессиях для протокола OSPF;

bgp – выводит информацию о BFD-сессиях для протокола BGP.

### Необходимый уровень привилегий

1

## Командный режим

ROOT

### Пример

```
esr# show bfd neighbors 192.168.39.230
Neighbor address:          192.168.39.230
Local address:            192.168.39.215
Interface:                gi1/0/3
Remote discriminator:     3427539848
Local discriminator:      3379089541
State:                    Up
Session type:             Control
Session mode:             Single-hop
Local diagnostic code:    No Diagnostic
Remote diagnostic code:  No Diagnostic
Minimal Tx Interval:     300 ms
Minimal Rx Interval:     300 ms
Multiplier:              5
Actual Tx Interval:      1000 ms
Actual Detection Interval: 5000 ms
Number of transmitted packets: 497
Number of received packets: 338
Uptime (d,h:m:s):       00,03:54:29
Client:                  BGP
Last received packet:
  Desired Min Tx Interval: 1000 ms
  Required Min Rx Interval: 1000 ms
  Multiplier:              5
```

## Настройка статических маршрутов IPv4/IPv6

- [ip bfd neighbor](#)
- [ip route](#)
- [ipv6 bfd neighbor](#)
- [ipv6 route](#)

### ip bfd neighbor

Данной командой запускается работа механизма BFD с определенным IP-адресом. Работа такого механизма необходима для удаления статических маршрутов из таблицы маршрутизации при недоступности next-hop.

Использование отрицательной формы команды (no) останавливает работу механизма BFD с определенным IP-адресом.

### Синтаксис

```
ip bfd neighbor <ADDR> [ { interface <IF> | tunnel <TUN> } ] [ local-address <ADDR>
[ multihop ] ] [ vrf <VRF> ]
no ip bfd neighbor <ADDR> [vrf <VRF>]
```

### Параметры

<ADDR> – IP-адрес шлюза, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IF> – интерфейс или группы интерфейсов, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<TUN> – имя туннеля, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#);

<VRF> – имя экземпляра VRF, задается строкой до 31 символа;

multihop – ключ для установки TTL=255, для работы механизма BFD через маршрутизируемую сеть.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# ip bfd neighbor 192.168.0.2
```

### ip route

Команда позволяет создать статический IP-маршрут к указанной подсети.

Использование отрицательной формы команды (no) удаляет указанный маршрут.

## Синтаксис

```
ip route [ vrf <VRF> ] <SUBNET> { { <NEXTHOP> [ resolve ] [ bfd ] [ unit <ID> ] |
interface <IF> | tunnel <TUN> | blackhole | unreachable | prohibit } [ track <TRACK-ID> ]
[ name <NAME>] } | wan load-balance rule <RULE> } [ <METRIC> ]
no ip route [ vrf <VRF> ] <SUBNET> [ <METRIC> ]
```

## Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа;

<SUBNET> – адрес назначения, может быть задан в следующих видах:

- AAA.BBB.CCC.DDD – IP-адрес хоста, где каждая часть принимает значения [0..255];
- AAA.BBB.CCC.DDD/NN – IP-адрес подсети с маской в виде префикса, где AAA-DDD принимают значения [0..255] и NN принимает значения [1..32].

<NEXTHOP> – IP-адрес шлюза задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

resolve – при указании данного параметра IP-адрес шлюза будет рекурсивно вычислен через таблицу маршрутизации. Если при рекурсивном вычислении не удастся найти шлюз из напрямую подключенной подсети, то данный маршрут будет отмечен как "Unreachable". Глубина поиска равна 5.

<ID> – номер юнита, принимает значения [1..2].

<IF> – имя IP-интерфейса, задается в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

<TUN> – имя туннеля, задается в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#).

<RULE> – номер правила wan, задается в диапазоне [1..50].

blackhole – при указании команды пакеты до данной подсети будут удаляться устройством без отправки уведомлений отправителю;

unreachable – при указании команды пакеты до данной подсети будут удаляться устройством, отправитель получит в ответ ICMP Destination unreachable (Host unreachable, code 1);

prohibit – при указании команды пакеты до данной подсети будут удаляться устройством, отправитель получит в ответ ICMP Destination unreachable (Communication administratively prohibited, code 13);

<METRIC> – метрика маршрута, принимает значения [0..255].

<TRACK-ID> – идентификатор Tracking-объекта. Если маршрут привязан к Tracking-объекту, то он появится в системе только при выполнении всех условий, заданных в объекте.

<NAME> – имя (описание) маршрута, текстовая переменная длиной до 31 символа.

bfd – при указании данного ключа активируется удаление статического маршрута в случае недоступности next-hop. Для работы данного механизма должен быть запущен механизм BFD с IP-адресом next-hop (см. команду **ip bfd neighbor** [выше](#)).

Проверка next-hop при помощи протокола bfd. В случае недоступности next-hop маршрут удаляется.

 Если в качестве подсети указать 0.0.0.0/0, то будет задан маршрут по умолчанию.

## Необходимый уровень привилегий

10

## Командный режим

### CONFIG

#### Пример 1

Задать маршрут до подсети 192.165.3.0/24 с метрикой 6 через шлюз 192.165.56.65:

```
esr(config)# ip route 192.165.3.0/24 192.165.56.65 6
```

#### Пример 2

Задать маршрут до подсети 192.165.3.0/24 с метрикой 6 через интерфейс GigabitEthernet 1/0/5:

```
esr(config)# ip route 192.165.3.0/24 interface gigabitethernet 1/0/5 6
```

#### Пример 3

Задать маршрут до подсети 192.165.3.0/24 через туннельный интерфейс vti 1:

```
esr(config)# ip route 192.165.3.0/24 interface vti 1
```

## ipv6 bfd neighbor

Данной командой запускается работа механизма BFD с определенным IPv6-адресом. Работа такого механизма необходима для удаления bgr-маршрутов из таблицы маршрутизации при недоступности next-hop.

Использование отрицательной формы команды (no) останавливает работу механизма BFD с определенным IPv6-адресом.

### Синтаксис

```
ipv6 bfd neighbor <IPV6-ADDR> [ interface <IF> ] [ local-address <IPV6-ADDR> ]  
[ multihop ] ] [ vrf <VRF> ]
```

```
no ipv6 bfd neighbor <IPV6-ADDR> [ vrf <VRF> ]
```

### Параметры

<IPV6-ADDR> – IPv6-адрес клиента, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<IF> – интерфейс или группы интерфейсов, задаётся в виде, описанном в разделе [Типы и порядок именованя интерфейсов маршрутизатора](#);

<VRF> – имя экземпляра VRF, задается строкой до 31 символа;

multihop – ключ для установки TTL=255, для работы механизма BFD через маршрутизируемую сеть.

### Необходимый уровень привилегий

10

## Командный режим

### CONFIG

#### Пример

```
esr(config)# ipv6 bfd neighbor FF02:0:0:0:1:FF00::/104
```

#### ipv6 route

Данная команда позволяет создать статический IPv6-маршрут к указанной подсети.

Использование отрицательной формы команды (no) удаляет указанный маршрут.

#### Синтаксис

```
ipv6 route [ vrf <VRF> ] <SUBNET> { { <NEXTHOP> [ resolve ] [ bfd ] [ unit <ID> ] |
interface <IF> | blackhole | unreachable | prohibit [ <METRIC> ] [ name <NAME> ] } | wan
load-balance rule <RULE> [ <METRIC> ] }
```

```
no ipv6 route [ vrf <VRF> ] <SUBNET> [ <METRIC> ]
```

#### Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа;

<SUBNET> – адрес назначения, может быть задан в следующих видах:

- X:X:X:X – IPv6-адрес хоста, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];
- X:X:X:X/EE – IPv6-адрес подсети с маской в виде префикса, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128].

<NEXTHOP> – IPv6-адрес шлюза, задаётся в виде X:X:X:X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

resolve – при указании данного параметра IPv6-адрес шлюза будет рекурсивно вычислен через таблицу маршрутизации. Если при рекурсивном вычислении не удастся найти шлюз из напрямую подключенной подсети, то данный маршрут не будет установлен в систему.

<ID> – номер юнита, принимает значения [1..2].

<IF> – имя IP-интерфейса, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

blackhole – при указании команды пакеты до данной подсети будут удаляться устройством без отправки уведомлений отправителю.


unreachable – при указании команды пакеты до данной подсети будут удаляться устройством, отправитель получит в ответ ICMP Destination unreachable (Host unreachable, code 1).

prohibit – при указании команды пакеты до данной подсети будут удаляться устройством, отправитель получит в ответ ICMP Destination unreachable (Communication administratively prohibited, code 13).

[METRIC] – метрика маршрута, принимает значения [0..255].

<NAME> – имя (описание) маршрута, текстовая переменная длиной до 31 символа.

bfd – при указании данного ключа, активируется проверка next-hop при помощи протокола bfd. В случае недоступности next-hop маршрут удаляется.

 Если в качестве подсети указать ::/0, то будет задан маршрут по умолчанию.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример 1

Задать маршрут до подсети 2001::/120 с метрикой 6 через шлюз fc00::1:

```
esr(config)# ipv6 route 2001::/120 fc00::1 6
```

### Пример 2

Задать маршрут до подсети 2001::/120 с метрикой 6 через интерфейс GigabitEthernet 1/0/5:

```
esr(config)# ipv6 route 2001::/120 interface gigabitethernet 1/0/5 6
```

### Пример 3

Задать маршрут до подсети 2001::/120 через туннельный интерфейс vti 1:

```
esr(config)# ipv6 route 2001::/120 interface vti 1
```



## Настройка протокола BGP

- address-family
- address-family
- advertise-map
- advertise-map
- aggregate-address
- allow-local-as
- as-range
- as-set
- attribute-map
- clear bgp
- cluster-id
- default-originate
- description
- ebgp-multihop
- fall-over
- flow-spec enable
- ipv6 router bgp log-neighbor-changes
- ipv6 router bgp maximum-paths
- listen-range
- local-as
- neighbor
- next-hop resolve-recursive
- next-hop-self
- no-prepend
- peer-group
- peer-group
- preference
- remote-as
- remove-private-as
- replace-as
- router bgp
- router bgp log-neighbor-changes
- router bgp maximum-paths
- route-reflector-client
- router-id
- send-community extended
- show bgp flow-spec
- show bgp ipv4 unicast
- show bgp ipv6 unicast
- show bgp l2vpn vpls
- show bgp neighbors
- show bgp summary
- show bgp vpnv4 unicast
- summary-only
- suppress-map
- timers error-wait
- timers holdtime
- timers keepalive
- update-source
- vrf
- weight

**address-family**

Данной командой осуществляется переход в режим настройки параметров передачи информации протоколов для процесса BGP.

Использование отрицательной формы команды (no) удаляет настроенные параметры передачи информации протоколов.

**Синтаксис**

```
[no] address-family { ipv4 | ipv6 } unicast
```

**Параметры**

- ipv4 – семейство ipv4;
- ipv6 – семейство ipv6.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-BGP

CONFIG-BGP-VRF

**Пример**

```
esr(config-bgp)# address-family ipv4
```

**address-family**

Данной командой определяется тип конфигурируемой маршрутной информации и переход в данный режим настройки.

Использование отрицательной формы команды (no) удаляет настроенные параметры передачи информации протоколов.

**Синтаксис**

```
[no] address-family { ipv4 | ipv6 | vpv4 | l2vpn vpls } unicast
```

**Параметры**

- ipv4 – семейство ipv4;
- ipv6 – семейство ipv6;
- vpv4 – семейство vpv4;
- l2vpn vpls – семейство l2vpn.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

**Пример**

```
esr(config-bgp-neighbor)# address-family ipv4
```

**advertise-map**

Данной командой активируется функция Conditional advertisement, позволяющая анонсировать маршрутную информацию, описанную в advertise-map, в зависимости от условий и наличия в BGP RIB-маршрутов, описанных в condition-map.

Использование отрицательной формы команды (no) отключает данную функцию.

**Синтаксис**

```
advertise-map <ADVERTISE> {EXIST-MAP | NOT-EXIST-MAP} <CONDITION>
no advertise-map
```

**Параметры**

<ADVERTISE> – имя конфигулируемых правил маршрутизации, задаётся строкой до 31 символа.

<CONDITION> – имя конфигулируемых правил маршрутизации, задаётся строкой до 31 символа.

<EXIST-MAP> – условие проверки: если маршруты, описанные в condition-map, присутствуют в BGP RIB, то происходит анонсирование маршрутов, описанных в advertise-map.

<NOT-EXIST-MAP> – условие проверки: если маршруты, описанные в condition-map, отсутствуют в BGP RIB, то происходит анонсирование маршрутов, описанных в advertise-map.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-BGP-NEIGHBOR-AF

CONFIG-BGP-GROUP-AF

CONFIG-BGP-NEIGHBOR-AF-VRF

CONFIG-BGP-GROUP-AF-VRF

**Пример**

```
esr(config-bgp-neighbor-af)# advertise-map ADVERTISE exist-map CONDITION
```

**advertise-map**

Данная команда позволяет задать дополнительные условия агрегирования маршрутов. Условия задаются через route-map.

Использование отрицательной формы команды (no) отключает данную функцию.

**Синтаксис**

```
advertise-map [ route-map <NAME> ]
no advertise-map
```

**Параметры**

<NAME> – имя маршрутной карты, задаётся строкой до 31 символа.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-BGP-AGGREGATE

CONFIG-BGP-VRF-AGGREGATE

**Пример**

```
esr(config-bgp-aggregate)# advertise-map example-route-map
```

**aggregate-address**

Данной командой включается функция агрегирования маршрутов, передаваемых по BGP.

Использование отрицательной формы команды (no) отключает данную функцию.

**Синтаксис**

```
[no] aggregate-address { <ADDR/LEN> | <IPV6-ADDR/LEN> }
```

**Параметры**

<ADDR/LEN> – IP-адрес и маска подсети, задается в виде AAA.BBB.CCC.DDD/EE, где AAA-DDD принимают значения [0..255] и EE принимает значения [1..32];

<IPV6-ADDR/LEN> – IPv6-адрес и маска подсети, задается в виде X:X:X::X/EE, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128].

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-BGP

## CONFIG-BGP-VRF

### Пример

```
esr(config-bgp)# aggregate-address 192.168.0.0/16
```

### allow-local-as

Данной командой задается режим, в котором разрешен приём маршрутов в BGP-атрибуте, AS Path которых содержатся номера автономной системы процесса.

Использование отрицательной формы команды (no) отключает данную функцию.

### Синтаксис

```
allow-local-as <NUMBER>  
no allow-local-as
```

### Параметры

<NUMBER> – пороговое число вхождений номера автономной системы процесса в атрибуте AS Path, при которых маршрут будет принят, диапазон допустимых значений [1..10].

### Необходимый уровень привилегий

10

### Командный режим

```
CONFIG-BGP-GROUP  
CONFIG-BGP-VRF-GROUP  
CONFIG-BGP-NEIGHBOR  
CONFIG-BGP-VRF-NEIGHBOR
```

### Пример

```
esr(config-bgp-neighbor)# allow-local-as
```

### as-range

Данная команда позволяет указать номера AS, с которыми будет динамически устанавливаться BGP-сессия.

Использование отрицательной формы команды (no) отключает данную функцию.

### Синтаксис

```
as-range <AS-PATH>  
no as-range <AS-PATH>
```

## Параметры

<AS-PATH> – список номеров автономных систем, задается в виде AS-AS,AS,AS-AS, принимает значения [1..4294967295].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-BGP-LISTEN

CONFIG-BGP-VRF-LISTEN

## Пример

```
esr(config-bgp-listen)# as-range 65000-65500,65527
esr(config-bgp-listen)#
```

## as-set

Данной командой задается режим, в котором в AS-Path агрегированного маршрута устанавливаются номера AS из AS-Path его компонентов.

Использование отрицательной формы команды (no) отключает данный режим.

## Синтаксис

[no] as-set

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-BGP-AGGREGATE

CONFIG-BGP-VRF-AGGREGATE

## Пример

```
esr(config-bgp-aggregate)# as-set
```

## attribute-map

Данная команда позволяет устанавливать дополнительные атрибуты агрегированного маршрута. Атрибуты указываются через route-map.

Использование отрицательной формы команды (no) отключает данную функцию.

### Синтаксис

```
attribute-map [ route-map <NAME> ]
no attribute-map
```

### Параметры

<NAME> – имя маршрутной карты, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-BGP-AGGREGATE

CONFIG-BGP-VRF-AGGREGATE

### Пример

```
esr(config-bgp-aggregate)# attribute-map example-route-map
```

### clear bgp

Данная команда сбрасывает все или определенный BGP-процесс.

### Синтаксис

```
clear bgp [ <AS> ] [ vrf <NAME_VRF> ] [ neighbor { <ADDR> | <IPV6_ADDR> } ]
```

### Параметры

<AS> – номер автономной системы, принимает значения [1..4294967295];

<ADDR> – IP-адрес соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6\_ADDR> – IPv6-адрес соседа, задается в виде X:X:X:X::X, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF];

<NAME-VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будут сброшены сессии протокола BGP.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

## Пример

```
esr# clear bgp
esr# clear bgp 1000
```

## cluster-id

Командой устанавливается идентификатор Route-Reflector кластера, которому принадлежит BGP-процесс маршрутизатора.

Использование отрицательной формы команды (no) удаляет идентификатор.

## Синтаксис

```
cluster-id <ID>
no cluster-id
```

## Параметры

<ID> – идентификатор Route-Reflector кластера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

## Необходимый уровень привилегий

10

## Командный режим

```
CONFIG-BGP
CONFIG-BGP-VRF
```

## Пример

```
esr(config-bgp-af)# cluster-id 1.1.1.1
```

## default-originate

Данной командой задается режим, в котором BGP-соседу в обновлении на ряду с другими маршрутами отправляется маршрут по умолчанию.

Использование отрицательной формы команды (no) отключает данную функцию.

## Синтаксис

```
[no] default-originate
```

## Параметры

Команда не содержит параметров.



**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR-FAMILY

CONFIG-BGP-VRF-NEIGHBOR-FAMILY

**Пример**

```
esr(config-bgp-neighbor)# default-originate
```

**description**

Данной командой определяется описание соседа.

Использование отрицательной формы команды (no) удаляет описание.

**Синтаксис**

```
description <DESCRIPTION>
```

```
no description
```

**Параметры**

<DESCRIPTION> – описание соседа, задаётся строкой до 255 символов.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

**Пример**

```
esr(config-bgp-neighbor)# description "ISP_RTK"
```

**ebgp-multihop**

Данной командой разрешается подключение к соседям, которые находятся в не напрямую подключенных подсетях.

Использование отрицательной формы команды (no) отключает данную функцию.

**Синтаксис**

```
[no] ebgp-multihop <NUM>
```

**Параметры**

<NUM> – максимальное количество хопов при установке EBGP (используется для TTL).

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-NEIGHBOR

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-NEIGHBOR

**Пример**

```
esr(config-bgp-neighbor)# ebgp-multihop
```

**fall-over**

Данная команда позволяет активировать один из методов обнаружения неактивного пиринга по протоколу BGP.

Использование отрицательной формы команды (no) отключает данную функцию.

**Синтаксис**

```
[no] fall-over { <BFD> | route-map <NAME> }
```

**Параметры**

<BFD> – активация протокола BFD на конфигурируемом BGP-соседе.

<NAME> – имя маршрутной карты, задаётся строкой до 31 символа. Данная настройка включает механизм [Fast Peer Deactivation](#).

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

## Пример

```
esr(config-bgp-neighbor)# fall-over bfd
```

## flow-spec enable

Данной командой устанавливается режим трансляции flow-spec информации при работе с BGP-соседом или группой BGP-соседей.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
[no] flow-spec enable
```

## Параметры

Отсутствуют.

## Значение по умолчанию

Отключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR-FAMILY

CONFIG-BGP-VRF-NEIGHBOR-FAMILY

## Пример

```
esr(config-bgp-group)# flow-spec enable
```

## ipv6 router bgp log-neighbor-changes

Данной командой включается логирование изменений состояния IPv6-BGP-соседей.

Использование отрицательной формы команды (no) отключает логирование изменений состояния IPv6-BGP-соседей.

## Синтаксис

```
[no] ipv6 router bgp log-neighbor-changes
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Отключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# ipv6 router bgp log-neighbor-changes
```

## ipv6 router bgp maximum-paths

Данной командой включается ECMP и определяется максимальное количество равноценных IPv6-маршрутов до цели.

Использование отрицательной формы команды (no) отключает ECMP.

## Синтаксис

```
ipv6 router bgp maximum-paths <VALUE>
```

```
no ipv6 router bgp maximum-paths
```

## Параметры

<VALUE> – количество допустимых равноценных IPv6-маршрутов до цели, принимает значения [1..16].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# ipv6 router bgp maximum-paths 14
```

## listen-range

Данная команда позволяет динамически устанавливать BGP-сессию с маршрутизаторами, которые имеют IP-адреса из указанной подсети, без указания конкретного адреса соседа.

Использование отрицательной формы команды (no) отключает данную функцию.

### Синтаксис

```
listen-range { <ADDR/LEN> | <IPV6-ADDR/LEN }
no listen-range { <ADDR/LEN> | <IPV6-ADDR/LEN }
```

### Параметры

<IP-ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IP-ADDR/LEN> – IP-подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32];

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-BGP

CONFIG-BGP-VRF

### Пример

```
esr(config-bgp)# listen-range 192.168.0.0/16
esr(config-bgp-listen)#
```

## local-as

Данная команда позволяет маршрутизатору представляться номером другой автономной системы в дополнение к своей реальной AS.

Использование отрицательной формы команды (no) отключает данную функцию.

### Синтаксис

```
local-as <AS>
no local-as
```

### Параметры

<AS> – номер автономной системы, принимает значения [1..4294967295].

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-VRF-NEIGHBOR

## Пример

```
esr(config-bgp-neighbor)# local-as 64500
esr(config-bgp-local-as)#
```

## neighbor

Данной командой добавляется BGP-сосед и осуществляется переход в режим настройки параметров BGP-соседа.

Использование отрицательной формы команды (no) удаляет параметры соседнего маршрутизатора из конфигурации.

## Синтаксис

```
[no] neighbor {<ADDR> |<IPV6-ADDR>}
```

## Параметры

<ADDR> – IP-адрес соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

<IPV6-ADDR> – IPv6-адрес клиента, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-BGP

CONFIG-BGP-VRF

## Пример

```
esr(config-bgp)# neighbor 192.168.0.2
esr(config-bgp-neighbor)#
```

## next-hop resolve-recursive

Данной командой устанавливается глубина рекурсивного поиска по next-hop атрибуту маршрута, полученного по протоколу BGP. Если поиск завершится безрезультатно, то маршрут будет помечен как "unreachable".

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
next-hop-resolve-recursive {<VALUE>}
no next-hop-resolve-recursive
```

**Параметры**

<VALUE> – глубина поиска, принимает значения [1..10].

**Значение по умолчанию**

1

**Необходимый уровень привилегий**

10

**Командный режим**

```
CONFIG-BGP-FAMILY
CONFIG-BGP-VRF-FAMILY
```

**Пример**

```
esr(config-bgp-af)# next-hop resolve-recursive 10
```

**next-hop-self**

Данной командой задается режим, в котором для всех маршрутов, полученных от eBGP-пиров, атрибут next-hop будет замен на адрес локального маршрутизатора при отправке iBGP-пирам.

Использование отрицательной формы команды (no) отключает данную функцию.

**Синтаксис**

```
next-hop-self [all]
no next-hop-self
```

**Параметры**

all – для eBGP- и iBGP-пиров в анонсируемой маршрутной информации в атрибуте next-hop будет указан адрес локального маршрутизатора.

**Необходимый уровень привилегий**

10

**Командный режим**

```
CONFIG-BGP-GROUP
CONFIG-BGP-GROUP-AF
```

CONFIG-BGP-GROUP-AF-VRF  
CONFIG-BGP-VRF-GROUP  
CONFIG-BGP-NEIGHBOR-AF  
CONFIG-BGP-NEIGHBOR-AF-VRF

### Пример

```
esr(config-bgp-neighbor-af)# next-hop-self
```

### no-prepend

Данная команда исключает добавление номера AS, указанного в опции local-as, в AS Path принимаемых маршрутов.

Использование отрицательной формы команды (no) отключает данную функцию.

### Синтаксис

[no] no-prepend

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-BGP-LOCAL-AS  
CONFIG-BGP-VRF-LOCAL-AS

### Пример

```
esr(config-bgp-local-as)# no-prepend
```

### peer-group

Данной командой создается группа BGP-соседей с заданным именем и осуществляется переход в режим конфигурирования параметров группы.

Использование отрицательной формы команды (no) удаляет BGP-группу с заданным именем.

### Синтаксис

[no] peer-group <NAME>

### Параметры

<NAME> – название группы, задается строкой до 31 символа.



**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-BGP

CONFIG-BGP-VRF

**Пример**

```
esr(config-bgp)# peer-group list1
```

**peer-group**

Данной командой к BGP-группе или BGP-соседу применяются настройки, описанные в заданной BGP-группе.

Использование отрицательной формы команды (no) отменяет применение настроек, описанных в заданной BGP-группе.

**Синтаксис**

```
peer-group <NAME>
```

```
no peer-group
```

**Параметры**

<NAME> – название группы, задается строкой до 31 символа.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

**Пример**

```
esr(config-bgp-af)# peer-group list1
```

**preference**

Данная команда определяет приоритетность маршрутов, получаемых от соседа.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
preference <VALUE>  
no preference
```

**Параметры**

<VALUE> – приоритетность маршрутов соседа, принимает значения в диапазоне [1..255].

**Значение по умолчанию**

170

**Необходимый уровень привилегий**

10

**Командный режим**

```
CONFIG-BGP-GROUP  
CONFIG-BGP-VRF-GROUP  
CONFIG-BGP-NEIGHBOR-FAMILY  
CONFIG-BGP-VRF-NEIGHBOR-FAMILY
```

**Пример**

```
esr(config-bgp-neighbor)# preference 30
```

**remote-as**

Данной командой устанавливается номер автономной системы BGP-соседа.  
Использование отрицательной формы команды (no) удаляет номер автономной системы.

**Синтаксис**

```
remote-as <AS>  
no remote-as
```

**Параметры**

<AS> – номер автономной системы, принимает значения [1..4294967295].

**Необходимый уровень привилегий**

10

**Командный режим**

```
CONFIG-BGP-GROUP  
CONFIG-BGP-VRF-GROUP
```

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

**Пример**

```
esr(config-bgp-neighbor)# remote-as 20
```

**remove-private-as**

Данной командой задается режим, в котором перед отправлением обновления из BGP-атрибута AS Path маршрутов удаляются приватные номера автономных систем (в соответствии с RFC 6996).

Использование отрицательной формы команды (no) отключает данную функцию.

**Синтаксис**

```
[no] remove-private-as <ACTION>
```

**Параметры**

<ACTION> – действие, указывающее на то, какие приватные AS должны быть удалены или заменены. Принимает одно из значений:

- all – удалить все включения приватных AS из оригинального AS path;
- nearest – удалить все приватные AS до последней (правой) публичной AS в оригинальном AS path;
- replace – заменить все приватные AS на номер AS, в процессе которого обрабатывается данная команда.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR-FAMILY

CONFIG-BGP-VRF-NEIGHBOR-FAMILY

**Пример**

```
esr(config-bgp-neighbor-af)# remove-private-as
```

**replace-as**

Данная команда исключает добавление в AS Path анонсируемого маршрута номера реальной AS.

Использование отрицательной формы команды (no) отключает данную функцию.

**Синтаксис**

```
[no] replace-as
```

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-BGP-LOCAL-AS

CONFIG-BGP-VRF-LOCAL-AS

## Пример

```
esr(config-bgp-local-as)# no-prepend
```

## router bgp

Данной командой добавляется BGP-процесс в систему и осуществляется переход в режим настройки параметров BGP-процесса.

Использование отрицательной формы команды (no) удаляет BGP-процесс из системы.

## Синтаксис

```
[no] router bgp <AS>
```

## Параметры

<AS> – номер автономной системы процесса, принимает значения [1..4294967295].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# router bgp 1000  
esr(config-bgp)#
```

Добавлен BGP-процесс с автономной системой 1000.

## router bgp log-neighbor-changes

Данной командой включается логирование изменений состояния BGP-соседей.

Использование отрицательной формы команды (no) отключает логирование изменений состояния BGP-соседей.

**Синтаксис**

```
[no] router bgp log-neighbor-changes
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# router bgp log-neighbor-changes
```

**router bgp maximum-paths**

Данной командой включается ECMP и определяется максимальное количество равноценных маршрутов до цели.

Использование отрицательной формы команды (no) отключает ECMP.

**Синтаксис**

```
router bgp maximum-paths <VALUE>  
no router bgp maximum-paths
```

**Параметры**

<VALUE> – количество допустимых равноценных маршрутов до цели, принимает значения [1..16].

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

## Пример

```
esr(config)# router bgp maximum-paths 14
```

## route-reflector-client

Данной командой указывается, что BGP-сосед является Route-Reflector клиентом.

Использование отрицательной формы команды (no) отключает данную функцию.

## Синтаксис

```
[no] route-reflector-client
```

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

## Пример

```
esr(config-bgp-neighbor)# route-reflector-client
```

## router-id

Данной командой устанавливается идентификатор маршрутизатора.

Использование отрицательной формы команды (no) удаляет идентификатор.

## Синтаксис

```
router-id { <ID> | <IF> | <TUN> }
```

```
no router-id
```

## Параметры

<ID> – идентификатор маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

<IF> – интерфейс, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#).

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-BGP

CONFIG-BGP-VRF

### Пример

```
esr(config-bgp-af)# router-id 1.1.1.1
```

### send-community extended

Включает отправку параметра extended community в отсылаемых соседу анонсах. По умолчанию параметры extended community удаляются из отправляемых анонсов. Для корректной работы VPNv4/L2VPN отправку extended community необходимо включить.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
[no] send-community extended
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

10

### Значение по умолчанию

Запрещено.

### Командный режим

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR-FAMILY

CONFIG-BGP-VRF-NEIGHBOR-FAMILY

### Пример

```
esr(config-bgp-neighbor-af)# send-community extended
```

## show bgp flow-spec

Данная команда отображает информацию о правилах flow-spec.

### Синтаксис

```
show bgp [ vrf <VRF-NAME> ] flow-spec [ neighbor <ADDR> ] [ instance <AS-NUM> ]
```

### Параметры

<AS-NUM> – номер автономной системы процесса, принимает значения [1..4294967295].

<ADDR> – IP-адрес, который задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

<VRF-NAME> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать протокол маршрутизации.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример 1

```

esr# show bgp flow-spec
-----
Destination Prefix: 192.168.54.2/32
IP Protocol:       17
Ext-community type: traffic-rate (0x8006)

```

## show bgp ipv4 unicast

Данная команда отображает таблицу маршрутизации BGP или детальную информацию об определенном маршруте при использовании фильтров.

### Синтаксис

```
show bgp [ vrf <VRF-NAME> ] ipv4 unicast [ { <IP-ADDR> | <IP-ADDR/LEN> } ] [ instance <AS-NUM> ] [ neighbor <IP-ADDR> [ { advertise-routes | routes } ] ]
```

### Параметры

<AS-NUM> – номер автономной системы процесса, принимает значения [1..4294967295];

<IP-ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IP-ADDR/LEN> – IP-подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32];

<VRF-NAME> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать протокол маршрутизации;

routes – при указании команды отображается маршрутная информация, полученная от соседа;



**advertise-routes** – при указании команды отображается маршрутная информация, объявленная соседу. Данная команда отображает таблицу маршрутизации BGP или детальную информацию об определенном маршруте при использовании фильтров.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример 1

```
esr# show bgp ipv4 unicast
Status codes: u - unicast, b - broadcast, m - multicast a - anycast
               * - valid, > - best
Origin codes: i - IGP, e - EGP, ? - incomplete
  Network          Next Hop          Metric  LocPrf  Path
*> u 10.0.10.0/24   10.115.0.1         100     100     i
*> u 0.0.0.0/0      10.115.0.1         100     100     i
* u 14.0.10.0/24   10.115.0.1         100     100     i
```

### Пример 2

```
esr# show bgp ipv4 unicast 75.0.0.0
75.0.0.0/24 via 115.0.0.40 on gi1/0/14 [bgp20 2000-01-15] (AS90?)
Administrative Distance: 68
Type: unicast
Origin: Incomplete
AS PATH: 1 30 70 90
Next Hop: 115.0.0.40
MED: 0
Local Preference: 100
Community: (1:555)
Valid, Best
```

### show bgp ipv6 unicast

Данная команда отображает таблицу маршрутизации BGP или детальную информацию об определенном маршруте при использовании фильтров.

### Синтаксис

```
show bgp [ vrf <VRF-NAME> ] ipv6 unicast [ { <IPV6-ADDR> | <IPV6-ADDR/LEN> } ] [ instance
<AS-NUM> ] [ neighbor <IPV6-ADDR> [ { advertise-routes | routes } ] ]
```

### Параметры

<AS-NUM> – номер автономной системы процесса, принимает значения [1..4294967295];

<IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<IPv6-ADDR/LEN> – подсеть, задаётся в виде X:X:X:X/EE, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF],EE принимает значения [1..128];

<VRF-NAME> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать протокол маршрутизации;

routes – при указании команды отображается маршрутная информация, полученная от соседа;

advertise-routes – при указании команды отображается маршрутная информация, объявленная соседу.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

### Пример 1

```
esr# show bgp ipv6 unicast
Status codes: u - unicast, b - broadcast, m - multicast, a - anycast
               * - valid, > - best
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network                Next Hop           Metric  LocPrf   Weight Path
*> u 2600:380:180::/41    fc00:0:1409:2800::1 --      100      0      65054 12389 3356 ...
i
*> u 2804:2404:8000::/34  fc00:0:1409:2800::1 --      100      0      65054 12389 266925 ..
. i
*> u 2400:3800:8800::/37  fc00:0:1409:2800::1 --      100      0      65054 12389 2497 ...
i
*> u 2400:54c0:c0::/44    fc00:0:1409:2800::1 --      100      0      65054 12389 3356 ...
i
```

### Пример 2

```
esr# show bgp ipv6 unicast 2600:380:180::/41
2600:380:180::/41 via fc00:0:1409:2800::1 on gi1/0/1.2800 [bgp65514 14:08:30] (20057i)
  Administrative Distance: 170
  Type:                    unicast
  Origin:                  IGP
  AS PATH:                 65054 12389 3356 7018 20057
  Next Hop:                fc00:0:1409:2800::1 (fe80::aaf9:4bff:feaa:2dd5)
  Local Preference:       100
  Community:               (3356:3) (3356:22) (3356:86) (3356:575) (3356:666) (3356:903) (3356:2059)
                          (12389:6)
  Weight:                  0
  Valid, Best
```

## show bgp l2vpn vpls

Данная команда отображает информацию о всех принятых l2vpn-маршрутах.

## Синтаксис

```
show bgp l2vpn vpls { all | rd <ASN:nn> } [ neighbor <ADDR> { advertise-routes | routes }
[ ve-id <ID> block-offset <ID> ]
```

## Параметры

- all – при указании команды all отображается полная информация l2vpn-маршрутов;
- rd – при указании команды rd отображается информация о маршрутах с заданным RD;
- routes – при указании команды отображается маршрутная информация, полученная от соседа;
- advertise-routes – при указании команды отображается маршрутная информация, объявленная соседу;
- <RD> – значение Route distinguisher, задается в одном из следующем виде:
  - <ASN>:<nn> – где <ASN> – принимает значение [1..65535], nn – принимает значение [1..65535];
  - <ADDR>:<nn> – где <ADDR> имеет вид – AAA.BBB.CCC.DDD/EE и AAA-DDD принимают значения [0..255], а nn – принимает значение [1..65535];
  - <4BASN>:<nn> – где <4ASN> – принимает значение [1..4294967295], nn – принимает значение [1..65535].
- <IP> – IP-адрес соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];
- <ID> – принимает значение [1..65535], задает идентификатор vpls edge или block-offset.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

### Блок кода

```
ESR-R# show bgp l2vpn vpls all
Status codes: * - valid, > - best, i - internal, S - stale
Origin codes: i - IGP, e - EGP, ? - incomplete
```

Codes	Route Distinguisher	VID	VBO	VBS	Next hop	Metric	LocPrf	Weight	Path
*>i i	65502:100	3	1	8	192.168.9.9	--	100	0	
*>	65502:102	2	1	10	--	--	--	--	
*>	65502:100	2	1	10	--	--	--	--	
*>i i	65502:102	1	1	10	192.168.7.7	--	100	0	
*>i i	65502:100	1	1	10	192.168.7.7	--	100	0	

## show bgp neighbors

Данная команда отображает информацию о всех или о выбранном BGP-пире.

## Синтаксис

```
show bgp [vrf <VRF-NAME>] neighbors [instance <AS-NUM> ] { <ADDR> | <IPV6-ADDR> }
```

## Параметры

<AS-NUM> – номер автономной системы процесса, принимает значения [1..4294967295].

<VRF-NAME> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать протокол маршрутизации.

<ADDR> – IP- или IPv6-адрес пира, который задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255], или в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

<IPV6-ADDR/LEN> – подсеть, задается в виде X:X:X::X/EE, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF], EE принимает значения [1..128].

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример 1

```
esr# show bgp instance 20
BGP neighbor is 10.115.0.1
  BGP state:           Established
  Neighbor address:    10.115.0.1
  Neighbor AS:         20
  Neighbor ID:         115.0.0.1
  Neighbor caps:       refresh restart-aware AS4
  Session:             internal multihop AS4
  Source address:      10.115.0.2
  Hold timer:          137/180
  Keepalive timer:     10/60
  Incoming prefix-list: from_ISP
  Outgoing prefix-list: to_ISP
  Incoming route-map:  comingS
  Outgoing route-map:  AS_prepend
  Uptime:              12 s
  BFD address:         192.168.1.2
  BFD state:           Up
  BFD interval:        3.000 s
  BFD timeout:         15.000 s
```

## Пример 2

```
esr# show bgp neighbors 100.64.28.1
BGP neighbor is 100.64.28.1
  BGP state:           Established
  Neighbor address:    100.64.28.1
  Neighbor AS:         65054
```

```

Neighbor ID:                5.5.0.111
Neighbor caps:              refresh enhanced-refresh restart-aware AS4
Session:                    external AS4
Source address:             100.64.28.160
Weight:                     0
Hold timer:                 165/180
Keepalive timer:           38/60
RR client:                  No
Address family ipv4 unicast:
  Default originate:        No
  Default information originate: No
  Outgoing route-map:      RouteMap_out
  Preference:               170
  Remove private AS:       No
  Next-hop self:            No
Uptime:                     12952 s

```

### Пример 3

```

esr# show bgp neighbors fc00:0:1409:2800::1
BGP neighbor is fc00:0:1409:2800::1
  BGP state:                Established
  Neighbor address:         fc00:0:1409:2800::1
  Neighbor AS:              65054
  Neighbor ID:              5.5.0.111
  Neighbor caps:            refresh enhanced-refresh restart-aware AS4
  Session:                  external AS4
  Source address:           fc00:0:1409:2800::16
  Weight:                   0
  Hold timer:               174/180
  Keepalive timer:         50/60
  RR client:                No
  Address family ipv6 unicast:
    Default originate:      No
    Default information originate: No
    Incoming route-map:     RouteMap_in
    Outgoing route-map:     RouteMap_out
    Preference:              170
    Remove private AS:       No
    Next-hop self:          No
  Uptime:                   13003 s

```

### show bgp summary

Данная команда отображает информацию о состоянии соединений с BGP-пирами.

#### Синтаксис

```
show bgp [vrf <VRF-NAME>] summary [instance <AS-NUM>]
```

#### Параметры

<AS-NUM> – номер автономной системы процесса, принимает значения [1..4294967295].

<VRF-NAME> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать протокол маршрутизации.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример**

```

esr# show bgp summary
Tue Sep 14 15:36:26 2021
  BGP router identifier 100.64.28.160, local AS number 65514
  BGP activity 984813/0 prefixes
Neighbor          AS                MsgRcvd      MsgSent      Up/Down      St/PfxRcd
-----
-----
-----
-----
-----
10.20.0.2         65058             0            0            00,01:28:02  Idle
100.64.28.1      65054            861119      102          00,01:27:56  855897

```

**show bgp vpv4 unicast**

Данная команда отображает информацию о vpv4-маршрутах. Указание префикса в команде отображает детальную информацию по префиксу.

**Синтаксис**

```

show bgp vpv4 unicast { all | rd <RD> | vrf <NAME_VRF> } [ neighbor <IP> { routes |
advertise-routes } ] [ <PREFIX> ]

```

**Параметры**

all – при указании команды all отображается полная информация vpv4-маршрутов;  
rd – при указании команды rd отображается информация о маршрутах с заданным RD;  
vrf – при указании команды vrf отображается информация vpv4-маршрутов для указанного VRF;  
routes – при указании команды routes отображается маршрутная информация, полученная от соседа;  
advertise-routes – при указании команды advertise-routes отображается маршрутная информация, объявленная соседу;  
<RD> – значение Route distinguisher, задается в одном из следующем виде:

- <ASN>:<nn> – где <ASN> – принимает значение [1..65535], nn – принимает значение [1..65535];
- <ADDR>:<nn> – где <ADDR> имеет вид – AAA.BBB.CCC.DDD/EE и AAA-DDD принимают значения [0..255], а nn – принимает значение [1..65535];
- <4BASN>:<nn> – где <4ASN> – принимает значение [1..4294967295], nn – принимает значение [1..65535];
- <NAME\_VRF> – имя экземпляра VRF, задается строкой до 31 символа.

<IP> – IP-адрес соседа, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<PREFIX> – префикс, имеет вид – AAA.BBB.CCC.DDD/EE, где AAA-DDD принимают значения [0..255] и EE принимает значения [1..32].

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример 1**

```

esr# show bgp vpnv4 unicast all
Status codes: * - valid, > - best, i - internal, S - stale
Origin codes: i - IGP, e - EGP, ? - incomplete

```

Codes Weight	Route Path	Distinguisher	IP Prefix	Next hop	Metric	Label	LocPrf
*>i 0	?	65000:10001	10.100.106.1/32	6.6.6.6	--	16	100
*> --	?	65000:10002	10.20.0.0/24	--	--	17	--
*> --	?	65000:10002	10.20.1.0/24	--	--	17	--
*>i 0	i	65000:10001	10.100.0.4/30	2.2.2.2	--	16	100
*> --	?	65000:10002	172.18.0.0/30	--	--	17	--
*>i 55	i	65000:10002	10.30.0.0/24	5.5.5.5	--	17	100
*>i 0	i	65000:10001	10.100.104.0/24	2.2.2.2	--	16	100
*> --	?	65000:10001	10.100.105.0/24	--	--	16	--
*>i 0	?	65000:10001	10.100.106.0/24	6.6.6.6	--	16	100
*> --	?	65000:10001	203.203.203.203/32	--	--	16	--
*> --	?	65000:10001	202.202.202.202/32	--	--	16	--
*>i 0	i	65000:10001	10.100.100.0/24	2.2.2.2	--	16	100
*> --	?	65000:10001	10.100.0.0/30	--	--	16	--
*> --	?	65000:10001	10.100.102.0/24	--	--	16	--

**Пример 2**

```

esr# show bgp vpnv4 unicast all 10.100.106.0/24
BGP routing table entry for 65000:10001 10.100.106.0/24
Next hop: 6.6.6.6
Label: 16
AS path: --
Origin: Incomplete
Local preference: 100
Extended Community: RT:65000:10001
Weight: 0

```

### summary-only

Данной командой задаётся режим, в котором будут подавляться все компоненты агрегированного маршрута.

Использование отрицательной формы команды (no) отключает данный режим.

### Синтаксис

```
[no] summary-only
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-BGP-AGGREGATE

CONFIG-BGP-VRF-AGGREGATE

### Пример

```
esr(config-bgp-aggregate)# summary-only
```

### suppress-map

Данная команда позволяет подавлять компоненты агрегированного маршрута. Параметры для подавления указываются через route-map.

Использование отрицательной формы команды (no) отключает данную функцию.

### Синтаксис

```
suppress-map [ route-map <NAME> ]
```

```
no suppress-map
```

### Параметры

<NAME> – имя маршрутной карты, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-BGP-AGGREGATE

CONFIG-BGP-VRF-AGGREGATE



## Пример

```
esr(config-bgp-aggregate)# suppress-map example-route-map
```

## timers error-wait

Данной командой устанавливается время минимальной и максимальной задержки, в течение которого запрещено устанавливать соединение, в целях защиты от частых разрывов соединения.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
timers error-wait <TIME1> <TIME2>
```

```
no timers error-wait
```

## Параметры

<TIME1> – время в секундах, принимает значения [1..65535];

<TIME2> – время в секундах, принимает значения [1..65535].

## Значение по умолчанию

60 и 300

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-BGP

CONFIG-VRF-BGP

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

## Пример

```
esr(config-bgp)# timers error-wait 90 450
```

## timers holdtime

Данной командой устанавливается временной интервал, по истечении которого встречная сторона считается недоступной. Таймер запускается после установления отношений соседства и начинает отсчёт от 0. Таймер сбрасывается при получении каждого ответа на keepalive-сообщение от встречной стороны. Рекомендуется устанавливать значение таймера равное  $3 * \textit{keepalive}$ .

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
timers holdtime <TIME>  
no timers holdtime
```

**Параметры**

<TIME> – время в секундах, принимает значения [2..65535].

**Значение по умолчанию**

180

**Необходимый уровень привилегий**

10

**Командный режим**

```
CONFIG-BGP  
CONFIG-VRF-BGP  
CONFIG-BGP-GROUP  
CONFIG-BGP-VRF-GROUP  
CONFIG-BGP-NEIGHBOR  
CONFIG-BGP-VRF-NEIGHBOR
```

**Пример**

```
esr(config-bgp-af)# timers holdtime 360
```

**timers keepalive**

Данной командой устанавливается временной интервал, по истечении которого идет проверка соединения со встречной стороной.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
timers keepalive <TIME>  
no timers keepalive
```

**Параметры**

<TIME> – время в секундах, принимает значения [1..32767].

**Значение по умолчанию**

60

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-BGP

CONFIG-VRF-BGP

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

**Пример**

```
esr(config-bgp-af)# timers keepalive 120
```

**update-source**

Данной командой определяется IP/IPv6-адрес маршрутизатора, который будет использоваться в качестве IP/IPv6-адреса источника в отправляемых обновлениях маршрутной информации BGP.

Использование отрицательной формы команды (no) удаляет указанный IP/IPv6-адрес источника.

**Синтаксис**

```
update-source { <ADDR> | <IPV6-ADDR> | <IF> | <TUN> }
```

```
no source-address
```

**Параметры**

<ADDR> – IP-адрес источника, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес источника, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<IF> – интерфейс, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#).

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

## CONFIG-BGP-VRF-NEIGHBOR

### Пример

```
esr(config-bgp-neighbor)# update-source 10.100.100.2
```

### vrf

Данной командой в системе создается экземпляр BGP процесса в VRF и осуществляется переход в режим настройки параметров BGP процесса в указанном VRF.

Использование отрицательной формы команды (no) удаляет экземпляр BGP VRF из системы.

### Синтаксис

```
[no] ip vrf <VRF>
```

### Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-BGP

### Пример

```
esr(config-bgp)# vrf xx  
esr(config-bgp-vrf)#
```

### weight

Данной командой устанавливается вес маршрутов, принимаемый от данного BGP-соседа или группы BGP-соседей.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
weight <WEIGHT>
```

```
no weight
```

### Параметры

<WEIGHT> – значение веса маршрута, принимает значения [0..65535].

### Значение по умолчанию

0

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

**Пример**

```
esr(config-bgp-neighbor)# weight 200
```

## Настройка протоколов RIP и RIPNG

- [authentication key-chain](#)
- [clear ip rip](#)
- [clear ipv6 rip](#)
- [ip rip metric](#)
- [ip rip mode](#)
- [ip rip neighbor](#)
- [ip rip summary-address](#)
- [ipv6 rip metric](#)
- [ipv6 rip summary-address](#)
- [ipv6 router rip](#)
- [passive-interface](#)
- [passive-interface](#)
- [router rip](#)
- [show ip rip](#)
- [show ipv6 rip](#)
- [timers flush](#)
- [timers invalid](#)
- [timers update](#)

### authentication key-chain

Данная команда определяет список паролей для аутентификации через алгоритм хеширования md5. Использование отрицательной формы команды (no) удаляет привязку к списку паролей.

### Синтаксис

```
authentication key-chain <KEYCHAIN>
no authentication key-chain
```

### Параметры

<KEYCHAIN> – идентификатор списка ключей, задаётся строкой до 16 символов.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-RIP

### Пример

```
esr(config-rip)# authentication key-chain lock
```

### clear ip rip

Данная команда удаляет содержимое базы маршрутов RIP.

**Синтаксис**

```
clear ip rip
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

10

**Командный режим**

ROOT

**Пример**

```
esr# clear ip rip
```

**clear ipv6 rip**

Данная команда удаляет содержимое базы маршрутов RIPNG.

**Синтаксис**

```
clear ipv6 rip
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

10

**Командный режим**

ROOT

**Пример**

```
esr# clear ipv6 rip
```

**ip rip metric**

Данная команда устанавливает величину метрики на интерфейсе.

Использование отрицательной формы команды (no) устанавливает значение метрики по умолчанию.

**Синтаксис**

```
ip rip metric <VALUE>  
no ip rip metric
```

**Параметры**

<VALUE> – величина метрики, задаётся в размере [1..15].

**Значение по умолчанию**

5

**Необходимый уровень привилегий**

10

**Командный режим**

```
CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-GRE  
CONFIG-IP4IP4  
CONFIG-LT
```

**Пример**

```
esr(config-if-gi)# ip rip metric 11
```

**ip rip mode**

Данная команда устанавливает режим анонсирования маршрутов.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
ip rip mode <MODE>  
no ip rip mode
```



## Параметры

<MODE> – режим анонсирования маршрутов:

- `multicast` – маршруты анонсируются в многоадресном режиме;
- `broadcast` – маршруты анонсируются в широковещательном режиме;
- `unicast` – маршруты анонсируются в unicast-режиме соседям, настроенным с помощью команды `ip rip neighbor <ADDR>`.

## Значение по умолчанию

`multicast`

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT

## Пример

```
esr(config-if-gi)# ip rip mode broadcast
```

## `ip rip neighbor`

Данной командой статически задается IP-адрес соседа для установления отношения в unicast-режиме анонсирования маршрутов.

Использование отрицательной формы команды (`no`) удаляет статически заданный адрес соседа.

## Синтаксис

```
[no] ip rip neighbor <ADDR>
```

**Параметры**

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT

**Пример**

```
esr(config-if-gi)# ip rip neighbor 10.100.100.5
```

**ip rip summary-address**

Данной командой включается суммаризация подсетей.

Использование отрицательной формы команды (no) отключает суммаризацию подсетей.

**Синтаксис**

```
[no] ip rip summary-address <ADDR/LEN>
```

**Параметры**

<ADDR/LEN> – IP-подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-GRE  
CONFIG-IP4IP4  
CONFIG-LT

### Пример

```
esr(config-if-gi)# ip rip summary-address 10.200.200.0/24
```

### ipv6 rip metric

Данная команда устанавливает величину метрики на интерфейсе.

Использование отрицательной формы команды (no) устанавливает значение метрики по умолчанию.

### Синтаксис

```
ipv6 rip metric <VALUE>  
no ipv6 rip metric
```

### Параметры

<VALUE> – величина метрики, задаётся в размере [1..15].

### Значение по умолчанию

5

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB

CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-GRE  
CONFIG-IP4IP4  
CONFIG-LT

### Пример

```
esr(config-if-gi)# ipv6 rip metric 11
```

### ipv6 rip summary-address

Данной командой включается суммаризация подсетей IPv6.

Использование отрицательной формы команды (no) отключает суммаризацию подсетей.

### Синтаксис

```
[no] ipv6 rip summary-address <ADDR/LEN>
```

### Параметры

<ADDR/LEN> – IP-подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-GRE  
CONFIG-IP4IP4

## CONFIG-LT

**Пример**

```
esr(config-if-gi)# ipv6 rip summary-address fb00::1/20
```

**ipv6 router rip**

Данной командой осуществляется переход в режим настройки параметров RIPNG-процесса.

Использование отрицательной формы команды (no) устанавливает параметры RIPNG-процесса на значения по умолчанию.

**Синтаксис**

```
[no] ipv6 router rip
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ipv6 router rip  
esr(config-ripng)#
```

**passive-interface**

Данная команда выключает анонсирования маршрутов по интерфейсу.

Использование отрицательной формы команды (no) восстанавливает анонсирования маршрутов.

**Синтаксис**

```
[no] passive-interface { <IF> | <TUN> }
```

**Параметры**

<IF> – интерфейс, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

<TUN> – имя туннеля, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#).

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-RIP

**Пример**

```
esr(config-rip)# passive-interface gigabitethernet 1/0/15
```

**passive-interface**

Данная команда выключает анонсирования маршрутов по интерфейсу.

Использование отрицательной формы команды (no) восстанавливает анонсирования маршрутов.

**Синтаксис**

```
[no] passive-interface <IF>
```

**Параметры**

<IF> – интерфейс, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-RIPNG

**Пример**

```
esr(config-ripng)# passive-interface gigabitethernet 1/0/15
```

**router rip**

Данной командой осуществляется переход в режим настройки параметров RIP-процесса.

Использование отрицательной формы команды (no) устанавливает параметры RIP-процесса на значения по умолчанию.

**Синтаксис**

```
[no] router rip
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Отключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# router rip
esr(config-rip)#
```

## show ip rip

Данная команда отображает таблицу маршрутизации RIP.

## Синтаксис

```
show ip rip
```

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
esr# show ip rip Sword# sho ip rip
10.10.0.1/32    via 115.0.0.10 on gi1/0/15 [rip 21:31:17] * (100/6)
10.1.90.0/24   via 115.0.0.10 on gi1/0/15 [rip 21:31:17] * (100/6)
192.168.16.0/24 via 115.0.0.10 on gi1/0/15 [rip 21:31:17] * (100/6)
```

## show ipv6 rip

Данная команда отображает таблицу маршрутизации RIPNG.

**Синтаксис**

```
show ipv6 rip
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример**

```
esr# show ipv6 rip Sword# sho ip rip
10.10.0.1/32    via 115.0.0.10 on gil/0/15 [rip 21:31:17] * (100/6)
10.1.90.0/24   via 115.0.0.10 on gil/0/15 [rip 21:31:17] * (100/6)
192.168.16.0/24 via 115.0.0.10 on gil/0/15 [rip 21:31:17] * (100/6)
```

**timers flush**

Данной командой устанавливается временной интервал, по истечении которого производится удаление маршрута. При установке значения нужно учитывать следующее правило: «timers invalid + 60».

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
timers flush <TIME>
```

```
no timers flush
```

**Параметры**

<TIME> – время в секундах, принимает значения [12..65535].

**Значение по умолчанию**

240 секунд.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-RIP

CONFIG-RIPNG



## Пример

```
esr(config-rip)# timers flush 300
```

### timers invalid

Данной командой определяется временной интервал корректности маршрутной записи без обновления. По истечении срока, без получения обновления, маршрут помечается как недоступный.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
timers invalid <TIME>
```

```
no timers invalid
```

### Параметры

<TIME> – время в секундах, принимает значения [12..65535].

### Значение по умолчанию

180

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-RIP

CONFIG-RIPNG

## Пример

```
esr(config-rip)# timers invalid 240
```

### timers update

Данной командой устанавливается временной интервал, по истечении которого производится анонсирование.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
timers update <TIME>
```

```
no timers update
```

**Параметры**

<TIME> – время в секундах, принимает значения [12..65535].

**Значение по умолчанию**

30

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-RIP

CONFIG-RIPNG

**Пример**

```
esr(config-rip)# timers update 25
```

## Настройка протоколов OSPF и OSPFv3

- [area](#)
- [area-type](#)
- [authentication key](#)
- [authentication key chain](#)
- [auto-cost reference-bandwidth](#)
- [default-cost](#)
- [default-information-originate](#)
- [default-metric-type](#)
- [clear ip ospf](#)
- [clear ipv6 ospf](#)
- [compatible rfc1583](#)
- [dead-interval](#)
- [hello-interval](#)
- [ip ospf](#)
- [ip ospf area](#)
- [ip ospf authentication algorithm](#)
- [ip ospf authentication key](#)
- [ip ospf authentication key-chain](#)
- [ip ospf bandwidth](#)
- [ip ospf bfd-enable](#)
- [ip ospf cost](#)
- [ip ospf dead-interval](#)
- [ip ospf hello-interval](#)
- [ip ospf instance](#)
- [ip ospf mtu-ignore](#)
- [ip ospf neighbor](#)
- [ip ospf network](#)
- [ip ospf passive-interface](#)
- [ip ospf poll-interval](#)
- [ip ospf priority](#)
- [ip ospf retransmit-interval](#)
- [ip ospf ttl-security-hops](#)
- [ip ospf wait-interval](#)
- [ipv6 ospf](#)
- [ipv6 ospf area](#)
- [ipv6 ospf bandwidth](#)
- [ipv6 ospf cost](#)
- [ipv6 ospf dead-interval](#)
- [ipv6 ospf hello-interval](#)
- [ipv6 ospf instance](#)
- [ipv6 ospf mtu-ignore](#)
- [ipv6 ospf neighbor](#)
- [ipv6 ospf network](#)
- [ipv6 ospf passive-interface](#)
- [ipv6 ospf poll-interval](#)
- [ipv6 ospf priority](#)
- [ipv6 ospf retransmit-interval](#)
- [ipv6 ospf ttl-security-hops](#)
- [ipv6 ospf wait-interval](#)
- [ipv6 router ospf](#)
- [ipv6 router ospf log-adjacency-changes](#)
- [maximum-paths](#)
- [opaque-lsa disable](#)
- [preference](#)

- [retransmit-interval](#)
- [router ospf](#)
- [router ospf log-adjacency-changes](#)
- [router-id](#)
- [show ip ospf](#)
- [show ip ospf database](#)
- [show ip ospf interface](#)
- [show ip ospf neighbors](#)
- [show ip ospf virtual-links](#)
- [show ipv6 ospf](#)
- [show ipv6 ospf database](#)
- [show ipv6 ospf interface](#)
- [show ipv6 ospf neighbors](#)
- [show ipv6 ospf virtual-links](#)
- [summary-address](#)
- [virtual-link](#)
- [wait-interval](#)

## area

Данной командой устанавливается идентификатор области.

Использование отрицательной формы команды (no) удаляет созданную область.

## Синтаксис

```
[no] area <AREA_ID>
```

## Параметры

<AREA\_ID> – идентификатор области, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-OSPF

CONFIG-OSPFV3

## Пример 1

```
esr(config-ospf)# area 11.11.11.51
```

## Пример 2

```
esr(config-ipv6-ospf)# area 11.11.11.51
```

## area-type

Команда определяет тип области.

Использование отрицательной формы команды (no) устанавливает тип области как стандартный.

### Синтаксис

```
[no] area-type <TYPE> [ no-summary ]
```

### Параметры

<TYPE> – тип области:

- normal – устанавливает значение standart (стандартная область);
- stub – устанавливает значение stub (тупиковая область).

no-summary – команда в связке с параметром «stub» образует область «totally stubby» (для передачи информации за пределы области используется только маршрут по умолчанию).

- nssa – устанавливает значение nssa (область NSSA).

no-summary – в связке с параметром nssa образует область totally nssa (автоматически генерирует маршрут по умолчанию как межобластной).

### Значение по умолчанию

normal

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-OSPF-AREA

CONFIG-OSPFV3-AREA

### Пример 1

```
esr(config-ospf-area)# area-type stub
```

### Пример 2

```
esr(config-ipv6-ospf-area)# area-type nssa
```

### authentication key

Данная команда определяет пароль для аутентификации через алгоритм хеширования md5 с соседом.

Использование отрицательной формы команды (no) удаляет пароль.

**Синтаксис**

```
authentication key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no authentication key
```

**Параметры**

<CLEAR-TEXT> – пароль, задаётся строкой от 1 до 8 символов;

<ENCRYPTED-TEXT> – зашифрованный пароль размером до 8 байт (от 2 до 16 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-OSPF-VLINK

**Пример**

```
esr(config-ospf-vlink)# authentication key ascii-text 12345678
```

**authentication key chain**

Данная команда определяет список паролей для аутентификации через алгоритм хеширования md5 с соседом.

Использование отрицательной формы команды (no) удаляет привязку к списку паролей.

**Синтаксис**

```
authentication key chain <KEYCHAIN>
no authentication key chain
```

**Параметры**

<KEYCHAIN> – идентификатор списка ключей, задаётся строкой до 16 символов.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-OSPF-VLINK

**Пример**

```
esr(config-ospf-vlink)# authentication key chain key2
```

## auto-cost reference-bandwidth

Команда задает референсное значение для расчета стоимости интерфейса (cost). Стоимость рассчитывается как отношение reference-bandwidth к bandwidth интерфейса. В случае если отношение меньше 1, стоимость принимается равной 1.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
auto-cost reference-bandwidth <VALUE>
no auto-cost reference-bandwidth
```

### Параметры

<VALUE> – величина референсного значения, задаётся в размере [1К..1000000000К]. Для удобства конфигурирования, допускается после численного значения вместо литеры 'К' (Kbps) использовать литеру 'М' (Mbps) или 'Г' (Gbps).

### Значение по умолчанию

100000К

### Необходимый уровень привилегий

10

### Командный режим

```
CONFIG-OSPF
CONFIG-OSPFV3
```

### Пример

```
esr(config-ospf)# auto-cost reference-bandwidth 100000К
```

## default-cost

Данная команда устанавливает стоимость маршрута по умолчанию, анонсированного в соответствующую зону (при включении "default-information-originate").

Использование отрицательной формы команды (no) возвращает стоимость маршрута в значение по умолчанию.

### Синтаксис

```
default-cost <VALUE>
no default-cost
```

### Параметры

<VALUE> – стоимость маршрута, принимает значения [1..65535].

**Значение по умолчанию**

1000

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-OSPF-AREA

CONFIG-OSPFV3-AREA

**Пример**

```
esr(config-ospf-area)# default-cost 100
```

**default-information-originate**

Данная команда позволяет включить анонсирование маршрута по умолчанию для соответствующей зоны. Применимо для следующих типов зон: Stub, Totally Stub, NSSA, Totally NSSA.

Для Stub, Totally Stub, Totally NSSA зон будет анонсирован маршрут типа IA (inter area).

Для NSSA возможно анонсирование маршрута типа external type-2 или type-1 (по умолчанию).

Использование отрицательной формы команды (no) выключает анонсирование маршрута по умолчанию.

**Синтаксис**

```
[no] default-information-originate
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-OSPF-AREA

CONFIG-OSPFV3-AREA

**Пример**

```
esr(config-ospf-area)# default-information-originate
```



**default-metric-type**

Данная команда устанавливает тип маршрута по умолчанию, анонсированного в соответствующей NSSA-зоне (при включении "default-information-originate").

Использование отрицательной формы команды (no) устанавливает тип маршрута в значение по умолчанию.

**Синтаксис**

```
default-metric-type { type-1 | type-2 }
no default-metric-type
```

**Параметры**

- type-1 – external-маршрут type-1 (E1);
- type-2 – external-маршрут type-2 (E2).

**Значение по умолчанию**

type-1

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-OSPF-AREA

CONFIG-OSPFV3-AREA

**Пример**

```
esr(config-ospf-area)# default-metric-type type-2
```

**clear ip ospf**

Данная команда сбрасывает все или определенный OSPF-процесс.

**Синтаксис**

```
clear ip ospf [ <ID> ] [ vrf <VRF> ]
```

**Параметры**

<ID> – номер процесса, принимает значения [1..65535].

<VRF> – имя экземпляра VRF, для которого будут сброшены все или определенный OSPF-процесс, задается строкой до 31 символа.

**Необходимый уровень привилегий**

10

## Командный режим

ROOT

### Пример

```
esr# clear ip ospf  
esr# clear ip ospf 1000
```

## clear ipv6 ospf

Данная команда сбрасывает все или определенный OSPFv3-процесс.

### Синтаксис

```
clear ipv6 ospf [ <ID> ] [ vrf <VRF> ]
```

### Параметры

<ID> – номер процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF, для которого будут сброшены все или определенный OSPFv3-процесс, задается строкой до 31 символа.

### Необходимый уровень привилегий

10

## Командный режим

ROOT

### Пример

```
esr# clear ipv6 ospf  
esr# clear ipv6 ospf 1000
```

## compatible rfc1583

Включается совместимость с RFC 1583.

Использование отрицательной формы команды (no) отключает совместимость с RFC 1583.

### Синтаксис

```
[no] compatible rfc1583
```

### Параметры

Команда не содержит параметров.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-OSPF

CONFIG-OSPFV3

**Пример**

```
esr(config-ospf)# compatible rfc1583
```

**dead-interval**

Данной командой устанавливается интервал времени в секундах, по истечении которого сосед будет считаться недоступным. Этот интервал должен быть кратным значению «hello-interval». Как правило, «dead-interval» равен 4 интервалам отправки hello-пакетов, то есть 40 секундам.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

**Синтаксис**

```
dead-interval <TIME>
```

```
no dead-interval
```

**Параметры**

<TIME> – время в секундах, принимает значения [2..65535].

**Значение по умолчанию**

40

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-OSPF-VLINK

CONFIG-OSPFV3-VLINK

**Пример**

```
esr(config-ospf-vlink)# dead-interval 60
```

## hello-interval

Данной командой устанавливается интервал времени в секундах, по истечении которого маршрутизатор отправляет следующий hello-пакет.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

### Синтаксис

```
hello-interval <TIME>
```

```
no hello-interval
```

### Параметры

<TIME> – время в секундах, принимает значения [1..65535].

### Значение по умолчанию

10

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-OSPF-VLINK

CONFIG-OSPFV3-VLINK

### Пример

```
esr(config-ospf-vlink)# hello-interval 8
```

## ip ospf

Данной командой включают маршрутизацию по протоколу OSPF на интерфейсе.

Использование отрицательной формы команды (no) отключает маршрутизацию по протоколу OSPF на интерфейсе.

### Синтаксис

```
[no] ip ospf
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-SERIAL  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IF-E1  
CONFIG-IF-MULTILINK  
CONFIG-IP4IP4  
CONFIG-VTI  
CONFIG-GRE  
CONFIG-LOOPBACK  
CONFIG-LT

## Пример

```
esr(config-if-gi)# ip ospf
```

## ip ospf area

Данная команда определяет принадлежность интерфейса к определенной области OSPF-процесса.

Использование отрицательной формы команды (no) удаляет принадлежность интерфейса к определенной области OSPF-процесса.

## Синтаксис

```
ip ospf area <AREA_ID>  
no ip ospf area
```

## Параметры

<AREA\_ID> – идентификатор области, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

## Пример

```
esr(config-ip4ip4)# ip ospf area 1.1.1.1
```

## ip ospf authentication algorithm

Данная команда определяет алгоритм аутентификации.

Использование отрицательной формы команды (no) отключает аутентификацию.

## Синтаксис

```
ip ospf authentication algorithm <ALGORITHM>
```

```
no ip ospf authentication algorithm
```

## Параметры

<ALGORITHM> – алгоритм аутентификации:

- cleartext – пароль, передается открытым текстом;
- md5 – пароль, хешируется по алгоритму md5.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-SERIAL  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IF-E1  
CONFIG-IF-MULTILINK  
CONFIG-IP4IP4  
CONFIG-VTI  
CONFIG-GRE  
CONFIG-LOOPBACK  
CONFIG-LT

## Пример

```
esr(config-if-gi)# ip ospf authentication algorithm cleartext
```

## ip ospf authentication key

Данная команда устанавливает пароль для аутентификации с соседом при передаче пароля открытым текстом.

Использование отрицательной формы команды (no) удаляет пароль.

## Синтаксис

```
ip ospf authentication key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }  
no ip ospf authentication key
```

## Параметры

<CLEAR-TEXT> – пароль, задаётся строкой от 1 до 8 символов;

<ENCRYPTED-TEXT> – зашифрованный пароль размером до 8 байт (от 2 до 16 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-SERIAL  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IF-E1  
CONFIG-IF-MULTILINK  
CONFIG-IP4IP4  
CONFIG-VTI  
CONFIG-GRE  
CONFIG-LOOPBACK  
CONFIG-LT

## Пример

```
esr(config-if-gi)# ip ospf authentication key ascii-text 123456789  
esr(config-if-gi)# ip ospf authentication key ascii-text encrypted CDE65039E5591FA3F1
```

## ip ospf authentication key-chain

Данная команда определяет список паролей для аутентификации через алгоритм хеширования md5 с соседом.

Использование отрицательной формы команды (no) удаляет привязку к списку паролей.

## Синтаксис

```
ip ospf authentication key-chain <KEYCHAIN>  
no ip ospf authentication key-chain
```

## Параметры

<KEYCHAIN> – идентификатор списка ключей, задаётся строкой до 16 символов.

## Необходимый уровень привилегий

15



## Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-SERIAL  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IF-E1  
CONFIG-IF-MULTILINK  
CONFIG-IP4IP4  
CONFIG-VTI  
CONFIG-GRE  
CONFIG-LOOPBACK  
CONFIG-LT

## Пример

```
esr(config-if-gi)# ip ospf authentication key-chain lock
```

## ip ospf bandwidth

Данная команда устанавливает величину пропускной способности интерфейса для автоматического расчета стоимости (cost).

Использование отрицательной формы команды (no) устанавливает значение метрики по умолчанию.

## Синтаксис

```
ip ospf bandwidth <VALUE>
```

```
no ip ospf bandwidth
```

## Параметры

<VALUE> – величина пропускной способности, задается в размере [1К..100000000К]. Для удобства конфигурирования допускается после численного значения вместо литеры 'К' (Kbps) использовать литеру 'М' (Mbps) или 'Г' (Gbps).

**Значение по умолчанию**

Тип интерфейса	Значение по умолчанию
Физический интерфейс	В соответствии с номинальной скоростью (1G для gigabitethernet 10G для tengigabitethernet и т. д.)
SUB и QinQ	Наследуется от родительского физического интерфейса
Bridge	1G
Туннель	100K
Loopback	80G
E1	1544K
Multilink	Определяется как сумма входящих в него интерфейсов, умноженная на E1 bandwidth
Port-channel	Определяется как сумма входящих в него интерфейсов, умноженная на номинальную скорость Port-channel

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

## Пример

```
esr(config-if-gi)# ip ospf bandwidth 100M
```

### ip ospf bfd-enable

Данной командой на интерфейсе включается протокол BFD для протокола OSPF.

Использование отрицательной формы команды (no) отключает на интерфейсе протокол BFD для протокола OSPF.

### Синтаксис

```
[no] ip ospf bfd-enable
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Протокол BFD для протокола OSPF отключен.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-LOOPBACK

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT

## Пример

```
esr(config-if-gi)# ip ospf bfd-enable
```

## ip ospf cost

Данная команда устанавливает величину метрики на интерфейсе или туннеле.

Использование отрицательной формы команды (no) устанавливает значение метрики, рассчитанное автоматически механизмом auto-cost значений ospf bandwidth и auto-cost reference-bandwidth.

## Синтаксис

```
ip ospf cost <VALUE>
```

```
no ip ospf cost
```

## Параметры

<VALUE> – величина метрики, задаётся в размере [0..32767].

## Значение по умолчанию

Рассчитывается автоматически исходя из значений ospf bandwidth и auto-cost reference-bandwidth.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-GRE

CONFIG-LOOPBACK

## CONFIG-LT

**Пример**

```
esr(config-if-gi)# ip ospf cost 11
```

**ip ospf dead-interval**

Данной командой устанавливается интервал времени в секундах, по истечении которого сосед будет считаться недоступным. Этот интервал должен быть кратным значению hello-interval. Как правило, dead-interval равен 4 интервалам отправки hello-пакетов, то есть 40 секундам.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

**Синтаксис**

```
ip ospf dead-interval <TIME>  
no ip ospf dead-interval
```

**Параметры**

<TIME> – время в секундах, принимает значения [2..65535].

**Значение по умолчанию**

40

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-GRE  
CONFIG-LOOPBACK  
CONFIG-LT

### Пример

```
esr(config-if-gi)# ip ospf dead-interval 60
```

### ip ospf hello-interval

Данной командой устанавливается интервал времени в секундах, по истечении которого маршрутизатор отправляет следующий hello-пакет.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

### Синтаксис

```
ip ospf hello-interval <TIME>  
no ip ospf hello-interval
```

### Параметры

<TIME> – время в секундах, принимает значения [1..255].

### Значение по умолчанию

10

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-SERIAL  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IF-E1  
CONFIG-IF-MULTILINK  
CONFIG-IP4IP4

CONFIG-VTI  
CONFIG-GRE  
CONFIG-LOOPBACK  
CONFIG-LT

### Пример

```
esr(config-if-gi)# ip ospf hello-interval 8
```

### ip ospf instance

Данная команда определяет принадлежность интерфейса к определенному OSPF-процессу.

Использование отрицательной формы команды (no) удаляет принадлежность интерфейса к OSPF-процессу.

### Синтаксис

```
ip ospf instance <ID>  
no ip ospf instance
```

### Параметры

<ID> – номер процесса, принимает значения [1..65535].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-SERIAL  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IF-E1  
CONFIG-IF-MULTILINK  
CONFIG-IP4IP4  
CONFIG-VTI  
CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

### Пример

```
esr(config-ip4ip4)# ip ospf instance 300
```

### ip ospf mtu-ignore

Данной командой включается режим, в котором OSPF-процесс будет игнорировать значение MTU интерфейса во входящих Database Description-пакетах.

Использование отрицательной формы команды (no) отключает режим игнорирования MTU интерфейса.

### Синтаксис

```
[no] ip ospf mtu-ignore
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT



## Пример

```
esr(config-if-gi)# ip ospf mtu-ignore
```

## ip ospf neighbor

Данной командой статически задается IP-адрес соседа для установления отношения в NBMA и P2MP (Point-to-MultiPoint) сетях. Использование отрицательной формы команды (no) удаляет статически заданный адрес соседа.

## Синтаксис

```
[no] ip ospf neighbor <IP> [ eligible ]
```

## Параметры

<IP> – IP-адрес соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

eligible – опциональный параметр, позволяет устройству участвовать в процессе выбора DR в NBMA-сетях. Приоритет интерфейса должен быть больше нуля, команда изменения приоритета описана в разделе [ip ospf priority](#).

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

## Пример

```
esr(config-if-gi)# ip ospf neighbor 10.0.0.2
```

## ip ospf network

Данная команда определяет тип сети.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
ip ospf network <TYPE>
```

```
no ip ospf network
```

## Параметры

<TYPE> – тип сети:

- broadcast – тип соединения широковещательный;
- non-broadcast – тип соединения NBMA;
- point-to-multipoint – тип соединения точка-многоточие;
- point-to-multipoint non-broadcast – тип соединения NBMA точка-многоточие;
- point-to-point – тип соединения точка-точка.

## Значение по умолчанию

broadcast

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-VTI  
CONFIG-GRE  
CONFIG-LOOPBACK  
CONFIG-LT

### Пример

```
esr(config-if-gi)# ip ospf network point-to-point
```

### ip ospf passive-interface

Данная команда переводит интерфейс в пассивный режим: с данного интерфейса не будут отправляться hello-пакеты, а также не будут устанавливаться отношения смежности (соседства), однако подсеть, подключенная к такому интерфейсу, будет анонсироваться другим участникам процесса. Использование отрицательной формы команды (no) возвращает интерфейс в режим работы с рассылкой hello-пакетов и установкой соседства.

### Синтаксис

```
[no] ip ospf passive-interface
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
CONFIG-TE  
CONFIG-TWE  
CONFIG-F0  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-SERIAL  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IF-E1  
CONFIG-IF-MULTILINK  
CONFIG-IP4IP4  
CONFIG-VTI  
CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

## Пример

```
esr(config-if-gi)# ip ospf passive-interface
```

## ip ospf poll-interval

Данная команда устанавливает интервал времени, в течение которого NBMA-интерфейс ждет, прежде чем отправить hello-пакет соседу, даже в случае, если сосед неактивен.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

## Синтаксис

```
ip ospf poll-interval <TIME>
```

```
no ip ospf poll-interval
```

## Параметры

<TIME> – время в секундах, принимает значения [1 .. 255].

## Значение по умолчанию

120

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-GRE  
CONFIG-LOOPBACK  
CONFIG-LT

### Пример

```
esr(config-if-gi)# ip ospf poll-interval 60
```

### ip ospf priority

Данной командой устанавливается приоритет маршрутизатора, который используется для выбора DR и BDR.

В качестве DR выбирается маршрутизатор с наибольшим приоритетом.

При равных приоритетах в качестве DR выбирается маршрутизатор с наибольшим router id.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ip ospf priority <VALUE>
```

```
no ip ospf priority
```

### Параметры

<VALUE> – приоритет интерфейса, принимает значения [0..255].

### Значение по умолчанию

128

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-SERIAL  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IF-E1

CONFIG-IF-MULTILINK  
CONFIG-IP4IP4  
CONFIG-VTI  
CONFIG-GRE  
CONFIG-LOOPBACK  
CONFIG-LT

### Пример

```
esr(config-if-gi)# ip ospf priority 200
```

### ip ospf retransmit-interval

Данной командой устанавливается интервал времени в секундах, по истечении которого маршрутизатор повторно отправит пакет, на который не получил подтверждения о получении (например, Database Description пакет или Link State Request пакеты).

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

### Синтаксис

```
ip ospf retransmit-interval <TIME>  
no ip ospf retransmit-interval
```

### Параметры

<TIME> – время в секундах, принимает значения [2..65535].

### Значение по умолчанию

5

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-SERIAL  
CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE  
CONFIG-IF-E1  
CONFIG-IF-MULTILINK  
CONFIG-IP4IP4  
CONFIG-VTI  
CONFIG-GRE  
CONFIG-LOOPBACK  
CONFIG-LT

### Пример

```
esr(config-if-gi)#ip ospf retransmit-interval 4
```

### ip ospf ttl-security-hops

Данной командой определяется значение параметра ttl в исходящих ospf-пакетах. Также во всех входящих ospf-пакетах будет осуществляться проверка, что значение ttl больше либо равно '255 - ttl-security-hops'.

Использование отрицательной формы команды (no) устанавливает значение ttl-security-hops по умолчанию.

### Синтаксис

```
ip ospf ttl-security-hops <VALUE>  
no ip ospf ttl-security-hops
```

### Параметры

<VALUE> – значение ttl [1..255].

### Значение по умолчанию

0

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
CONFIG-TE  
CONFIG-TWE  
CONFIG-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ

CONFIG-SERIAL  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IP4IP4  
CONFIG-VTI  
CONFIG-GRE  
CONFIG-LOOPBACK  
CONFIG-LT  
CONFIG-IF-MULTILINK

### Пример

```
esr(config-if-gi)#ip ospf ttl-security-hops 200
```

### ip ospf wait-interval

Данной командой определяется интервал времени в секундах, по истечении которого маршрутизатор выберет DR в сети.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

### Синтаксис

```
ip ospf wait-interval <TIME>  
no ip ospf wait-interval
```

### Параметры

<TIME> – время в секундах, принимает значения [2..65535].

### Значение по умолчанию

40

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ



CONFIG-SERIAL  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IP4IP4  
CONFIG-VTI  
CONFIG-GRE  
CONFIG-LOOPBACK  
CONFIG-LT

### Пример

```
esr(config-if-gi)# ip ospf wait-interval 60
```

### ipv6 ospf

Данной командой включается маршрутизация по протоколу OSPFv3 на интерфейсе.

Использование отрицательной формы команды (no) отключает маршрутизацию по протоколу OSPFv3 на интерфейсе.

### Синтаксис

[no] ipv6 ospf

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-LOOPBACK  
CONFIG-LT

## Пример

```
esr(config-if-gi)# ipv6 ospf
```

### ipv6 ospf area

Данная команда определяет принадлежность интерфейса к определенной области OSPFv3-процесса.

Использование отрицательной формы команды (no) удаляет принадлежность интерфейса к определенной области OSPFv3-процесса.

### Синтаксис

```
ipv6 ospf area <AREA_ID>
```

```
no ipv6 ospf area
```

### Параметры

<AREA\_ID> – идентификатор области, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

## Пример

```
esr(config-ip4ip4)# ipv6 ospf area 1.1.1.1
```

### ipv6 ospf bandwidth

Данная команда устанавливает величину пропускной способности интерфейса для автоматического расчета стоимости (cost).

Использование отрицательной формы команды (no) устанавливает значение метрики по умолчанию.

## Синтаксис

```
ipv6 ospf bandwidth <VALUE>
```

```
ipv6 ip ospf bandwidth
```

## Параметры

<VALUE> – величина пропускной способности, задаётся в размере [1K..100000000K]. Для удобства конфигурирования допускается после численного значения вместо литеры 'K' (Kbps) использовать литеру 'M' (Mbps) или 'G' (Gbps).

## Значение по умолчанию

Тип интерфейса	Значение по умолчанию
Физический интерфейс	В соответствии с номинальной скоростью (1G для gigabitethernet, 10G для tengigabitethernet и т. д.)
SUB и QinQ	Наследуется от родительского физического интерфейса
Bridge	1G
Туннель	100K
Loopback	80G
E1	1544K
Multilink	Определяется как сумма входящих в него интерфейсов, умноженная на E1 bandwidth
Port-channel	Определяется как сумма входящих в него интерфейсов, умноженная на номинальную скорость Port-channel

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE  
CONFIG-IF-E1  
CONFIG-IF-MULTILINK  
CONFIG-IP4IP4  
CONFIG-VTI  
CONFIG-GRE  
CONFIG-LOOPBACK  
CONFIG-LT

### Пример

```
esr(config-if-gi)# ipv6 ospf bandwidth 100M
```

### ipv6 ospf cost

Данная команда устанавливает величину метрики на интерфейсе или туннеле.

Использование отрицательной формы команды (no) устанавливает значение метрики по умолчанию.

### Синтаксис

```
ipv6 ospf cost <VALUE>
```

```
no ipv6 ospf cost
```

### Параметры

<VALUE> – величина метрики, задаётся в размере [1..65535].

### Значение по умолчанию

Рассчитывается автоматически исходя из значений ospf bandwidth и auto-cost reference-bandwidth.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

### Пример

```
esr(config-if-gi)# ipv6 ospf cost 11
```

### ipv6 ospf dead-interval

Данной командой устанавливается интервал времени в секундах, по истечении которого сосед будет считаться недоступным. Этот интервал должен быть кратным значению hello-interval. Как правило, dead-interval равен 4 интервалам отправки hello-пакетов, то есть 40 секундам.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

### Синтаксис

```
ipv6 ospf dead-interval <TIME>
```

```
no ipv6 ospf dead-interval
```

### Параметры

<TIME> – время в секундах, принимает значения [2..65535].

### Значение по умолчанию

40

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

## Пример

```
esr(config-if-gi)# ipv6 ospf dead-interval 60
```

## ipv6 ospf hello-interval

Данной командой устанавливается интервал времени в секундах, по истечении которого маршрутизатор отправляет следующий hello-пакет.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

## Синтаксис

```
ipv6 ospf hello-interval <TIME>
```

```
no ipv6 ospf hello-interval
```

## Параметры

<TIME> – время в секундах, принимает значения [1..255].

## Значение по умолчанию

10

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

## Пример

```
esr(config-if-gi)# ipv6 ospf hello-interval 8
```

## ipv6 ospf instance

Данная команда определяет принадлежность интерфейса к определенному OSPFv3-процессу.

Использование отрицательной формы команды (no) удаляет принадлежность интерфейса к OSPFv3-процессу.

### Синтаксис

```
ipv6 ospf instance <ID>  
no ipv6 ospf instance
```

### Параметры

<ID> – номер процесса, принимает значения [1..65535].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-LOOPBACK  
CONFIG-LT

### Пример

```
esr(config-ip4ip4)# ipv6 ospf instance 300
```

## ipv6 ospf mtu-ignore

Данной командой включается режим, в котором OSPFv3-процесс будет игнорировать значение MTU интерфейса во входящих Database Description пакетах.

Использование отрицательной формы команды (no) отключает режим игнорирования MTU интерфейса.

### Синтаксис

```
[no] ipv6 ospf mtu-ignore
```

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

## Пример

```
esr(config-if-gi)# ipv6 ospf mtu-ignore
```

## ipv6 ospf neighbor

Данной командой статически задается IPv6-адрес соседа для установления отношения в NMBA и P2MP (Point-to-MultiPoint) сетях. Использование отрицательной формы команды (no) удаляет статически заданный адрес соседа.

## Синтаксис

```
[no] ipv6 ospf neighbor <IPV6-ADDR> [ eligible ]
```

## Параметры

<IPV6-ADDR> – IPv6-адрес соседа, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

eligible – опциональный параметр, позволяет устройству участвовать в процессе выбора DR в NBMA-сетях. Приоритет интерфейса должен быть больше нуля, команда изменения приоритета описана в разделе [ip ospf priority](#).

## Необходимый уровень привилегий

10



## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

## Пример

```
esr(config-if-gi)# ipv6 ospf neighbor fc00::2
```

## ipv6 ospf network

Данная команда определяет тип сети.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
ipv6 ospf network <TYPE>
```

```
no ipv6 ospf network
```

## Параметры

<TYPE> – тип сети:

- broadcast – тип соединения широковещательный;
- non-broadcast – тип соединения NBMA;
- point-to-multipoint – тип соединения точка-многоточие;
- point-to-multipoint non-broadcast – тип соединения NBMA точка-многоточие;
- point-to-point – тип соединения точка-точка.

## Значение по умолчанию

broadcast

## Необходимый уровень привилегий

10

**Командный режим**

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-LOOPBACK  
CONFIG-LT

**Пример**

```
esr(config-if-gi)# ipv6 ospf network point-to-point
```

**ipv6 ospf passive-interface**

Данная команда переводит интерфейс в пассивный режим, что означает что с данного интерфейса не будут отправляться hello-пакеты а также не будут устанавливаться отношения смежности (соседства), однако подсеть, подключенная к такому интерфейсу, будет анонсироваться другим участникам процесса. Использование отрицательной формы команды (no) возвращает интерфейс в режим работы с рассылкой hello-пакетов и установкой соседства.

**Синтаксис**

```
[no] ipv6 ospf passive-interface
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB

CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-LOOPBACK  
CONFIG-LT

### Пример

```
esr(config-if-twe)# ipv6 ospf passive-interface
```

### ipv6 ospf poll-interval

Данная команда устанавливает интервал времени, в течение которого NBMA-интерфейс ждет, прежде чем отправить hello-пакет соседу, даже в случае, если сосед неактивен.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

### Синтаксис

```
ipv6 ospf poll-interval <TIME>  
no ipv6 ospf poll-interval
```

### Параметры

<TIME> – время в секундах, принимает значения [1..255].

### Значение по умолчанию

120 секунд

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-LOOPBACK  
CONFIG-LT

## Пример

```
esr(config-if-gi)# ipv6 ospf poll-interval 60
```

## ipv6 ospf priority

Данной командой устанавливается приоритет маршрутизатора, который используется для выбора DR и BDR.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
ipv6 ospf priority <VALUE>
```

```
no ipv6 ospf priority
```

## Параметры

<VALUE> – приоритет интерфейса, принимает значения [0..255].

## Значение по умолчанию

120

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

## Пример

```
esr(config-if-gi)# ipv6 ospf priority 300
```

## ipv6 ospf retransmit-interval

Данной командой устанавливается интервал времени в секундах, по истечении которого маршрутизатор повторно отправит пакет, на который не получил подтверждения о получении (например, Database Description пакет или Link State Request пакеты).

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

### Синтаксис

```
ipv6 ospf retransmit-interval <TIME>  
no ipv6 ospf retransmit-interval
```

### Параметры

<TIME> – время в секундах, принимает значения [2..65535].

### Значение по умолчанию

5

### Необходимый уровень привилегий

10

### Командный режим

```
CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-LOOPBACK  
CONFIG-LT
```

### Пример

```
esr(config-if-gi)#ipv6 ospf retransmit-interval 4
```

## ipv6 ospf ttl-security-hops

Данной командой определяется значение параметра ttl в исходящих ospf-пакетах. Также во всех входящих ospf-пакетах будет осуществляться проверка, что значение ttl больше либо равно '255 - ttl-security-hops'.

Использование отрицательной формы команды (no) устанавливает значение ttl-security-hops по умолчанию.

### Синтаксис

```
ipv6 ospf ttl-security-hops <VALUE>  
no ipv6 ospf ttl-security-hops
```

### Параметры

<VALUE> – значение ttl [1..255].

### Значение по умолчанию

0

### Необходимый уровень привилегий

10

### Командный режим

```
CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-SERIAL  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IP4IP4  
CONFIG-VTI  
CONFIG-GRE  
CONFIG-LOOPBACK  
CONFIG-LT
```

### Пример

```
esr(config-if-gi)#ipv6 ospf ttl-security-hops 200
```

### ipv6 ospf wait-interval

Данной командой определяется интервал времени в секундах, по истечении которого маршрутизатор выберет DR в сети.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

### Синтаксис

```
ipv6 ospf wait-interval <TIME>  
no ipv6 ospf wait-interval
```

### Параметры

<TIME> – время в секундах, принимает значения [2..65535].

### Значение по умолчанию

40 секунд

### Необходимый уровень привилегий

10

### Командный режим

```
CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-LOOPBACK  
CONFIG-LT
```

### Пример

```
esr(config-if-gi)# ipv6 ospf wait-interval 60
```

### ipv6 router ospf

Командой добавляется OSPFv3-процесс в систему и осуществляется переход в режим настройки параметров OSPFv3-процесса.

Использование отрицательной формы команды (no) удаляет OSPFv3-процесс из системы.

### Синтаксис

```
[no] ipv6 router ospf <ID> [vrf <VRF>]
```

## Параметры

<ID> – номер автономной системы процесса, принимает значения [1..65535].

<VRF> – имя экземпляра VRF, в котором будет работать OSPFv3-процесс, задается строкой до 31 символа.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# ipv6 router ospf 300
esr(config-ipv6-ospf)#
```

## ipv6 router ospf log-adjacency-changes

Данная команда позволяет включить вывод информации о состоянии отношений с соседями для протокола маршрутизации OSPFv3.

Использование отрицательной формы команды (no) отключает вывод информации.

## Синтаксис

[no] ipv6 router ospf log-adjacency-changes

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# ipv6 router ospf log-adjacency-changes
```

## maximum-paths

Данной командой определяется максимальное количество равноценных маршрутов до цели.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.



## Синтаксис

```
maximum-paths <PATHS>  
no maximum-paths
```

## Параметры

<PATHS> – количество допустимых равноценных маршрутов до цели, принимает значения [1..32].

## Значение по умолчанию

16

## Необходимый уровень привилегий

10

## Командный режим

```
CONFIG-OSPF  
CONFIG-OSPFV3
```

## Пример 1

```
esr(config-ospf)# maximum-paths 10
```


## Пример 2

```
esr(config-ipv6-ospf)# maximum-paths 10
```

## opaque-lsa disable

Данной командой отключается поддержка OPAQUE LSA сообщений для текущего процесса OSPF. В результате в процессе установки смежности маршрутизатор сообщит об отсутствии поддержки всех типов OPAQUE LSA.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

 После отключения поддержки OPAQUE LSA функционал, использующий данный тип LSA (например, Graceful-restart), будет деактивирован.

## Синтаксис

```
[no] opaque-lsa disable
```

## Параметры

Отсутствуют.

**Значение по умолчанию**

Поддержка OPAQUE LSA включена.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-OSPF

**Пример**

```
esr(config-ospf)# opaque-lsa disable
```

**preference**

Данная команда определяет приоритетность маршрутов процесса OSPF/OSPFv3.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
preference <VALUE>
```

```
no preference
```

**Параметры**

<VALUE> – приоритетность маршрутов процесса OSPF, принимает значения в диапазоне [1..255].

**Значение по умолчанию**

Наследуется от глобального значения preference, устанавливаемого командой [ip protocols preference](#).

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-OSPF

CONFIG-OSPFV3

**Пример 1**

```
esr(config-ospf)# preference 30
```

## Пример 2

```
esr(config-ipv6-ospf)# preference 30
```

### retransmit-interval

Данной командой устанавливается интервал времени в секундах, по истечении которого маршрутизатор повторно отправит пакет, который не получил подтверждение о получении (например, Database Description пакет или Link State Request пакеты).

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

### Синтаксис

```
retransmit-interval <TIME>
```

```
no retransmit-interval
```

### Параметры

<TIME> – время в секундах, принимает значения [2..65535].

### Значение по умолчанию

5

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-OSPF-VLINK

CONFIG-OSPFV3-VLINK

### Пример

```
esr(config-ospf-vlink)# retransmit-interval 4
```

### router ospf

Данной командой добавляется OSPF-процесс в систему и осуществляется переход в режим настройки параметров OSPF-процесса.

Использование отрицательной формы команды (no) удаляет OSPF-процесс из системы.

### Синтаксис

```
[no] router ospf <ID> [vrf <VRF>]
```

## Параметры

<ID> – номер автономной системы процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать протокол маршрутизации.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# router ospf 300  
esr(config-ospf)#
```

## router ospf log-adjacency-changes

Данная команда позволяет включить вывод информации о состоянии отношений с соседями для протокола маршрутизации OSPFv2.

Использование отрицательной формы команды (no) отключает вывод информации.

## Синтаксис

[no] router ospf log-adjacency-changes

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Отключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# router bgp log-adjacency-changes
```

## router-id

Данной командой устанавливается идентификатор маршрутизатора.

Использование отрицательной формы команды (no) удаляет идентификатор.

### Синтаксис

```
router-id { <ID> | <IF> | <TUN> }  
no router-id
```

### Параметры

<ID> – идентификатор маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IF> – интерфейс, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#).

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-OSPF

CONFIG-OSPFV3

### Пример

```
esr(config-ospf)# router-id 1.1.1.1
```

## show ip ospf

Данная команда отображает таблицу маршрутизации OSPF, если не указан аргумент. При указании процесса выводит информацию о конфигурации интерфейсов по данному процессу.

### Синтаксис

```
show ip ospf [ <ID> ] [ vrf <VRF> ]
```

### Параметры

<ID> – номер процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF, для которого будет отображена таблица маршрутизации OSPF, задается строкой до 31 символа.

### Необходимый уровень привилегий

1

## Командный режим

ROOT

### Пример

Отображение таблицы маршрутизации:

```

esr# show ip ospf
0      2.2.2.0/24      [150/10] dev gi1/0/1      [ospf2 19:40:31] (2.2.2.2)

```

### show ip ospf database

Данная команда отображает базу данных OSPF.

### Синтаксис

```

show ip ospf <ID> [ vrf <VRF> ] database [ active ] [ area-id <AREA-ID> ] [ detailed ]
[ { asbr-summary | external | network | nssa-external | opaque-link | router |
summary } ] [ lsa-id <LSA-ID> ] [ router-id <ROUTER-ID> ] [ self-originate ]
[ statistic ]

```

### Параметры

active – показать LSA, время жизни которых менее 3600 секунд;

<AREA-ID> – идентификатор области, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

asbr-summary – показать LSA Type 4 – ASBR-summary-LSA;

detailed – показать расширенную информацию о LSA;

external – показать LSA Type 5 – AS-external-LSA;

<LSA-ID> – идентификатор LSA, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

network – показать LSA Type 2 – Network-LSA;

nssa-external – показать LSA Type 7 – NSSA-external-LSA;

opaque-link – показать Opaque LSA;

router – показать LSA Type 1 – Router-LSA;

<ROUTER-ID> – идентификатор маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

self-originate – показать локально сгенерированные LSA;

statistic – показать статистическую информацию о LSA, хранящихся в LSDB;

summary – показать LSA Type 3 – Summary-LSA;

<ID> – номер OSPF-процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF для OSPF-процесса, в котором будет отображена таблица данных, задается строкой до 31 символа.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример**

```

esr# show ip ospf 111 dababase
Global
Type   LS ID           Router           Age  Sequence  Checksum
0005   10.166.11.12   10.1.0.1        1020 80000013   01b7
0005   0.0.0.0        10.166.11.1     245  80000010   aa48
0005   10.62.19.128   10.166.11.1     725  8000000e   6d2b
0005   10.62.20.0     10.166.11.1     731  8000000d   69af
0005   10.62.20.128   10.166.11.1     244  80000010   5e37
0005   10.62.21.128   10.166.11.1     244  80000010   5341
0005   10.166.11.0    10.166.11.1     245  80000010   cc6d
0005   10.166.11.12   10.166.11.1     245  80000010   54d9
Area 0.0.11.1
Type   LS ID           Router           Age  Sequence  Checksum
0001   10.1.0.1       10.1.0.1        1015 80000067   989e
0001   10.166.11.1    10.166.11.1     1021 80000018   8d96
0002   10.166.11.14   10.166.11.1     1021 80000001   68a5

```

**show ip ospf interface**

Данная команда отображает информацию об OSPF-интерфейсе.

**Синтаксис**

```
show ip ospf interface [ vrf <VRF> ] [ <IF> | <TUN> ]
```

**Параметры**

<IF> – интерфейс или группы интерфейсов, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<TUN> – имя туннеля, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#);

<VRF> – имя экземпляра VRF для OSPF-процесса, в котором будет отображена информация об OSPF-интерфейсе, задается строкой до 31 символа.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

## Пример

```

esr# show ip ospf interface gi1/0/1
Interface:                gigabitethernet 1/0/1
Internet Address:         25.25.0.1/24
Router ID:                 6.0.0.1
Network Type:             broadcast
Area:                     0.0.0.0 (0)
Interface has:            no authentication
Transmit:                 1
State:                    dr
Priority:                  128
Cost:                     10
ECMP weight:              1
Hello timer:              10
Wait timer:               40
Dead timer:               40
Retransmit timer:        5
Designed router (ID):     6.0.0.1
Designed router (IP):     25.25.0.1
Backup designed router (ID): 6.0.0.3
Backup designed router (IP): 25.25.0.3
Neighbor Count:          0
Adjacent neighbor count: 0

```

## show ip ospf neighbors

Данная команда отображает информацию о всех соседях или соседях определенного OSPF-процесса.

### Синтаксис

```

show ip ospf [ <ID> [ vrf <VRF> ] ] neighbors
show ip ospf neighbors[ <ID> [ vrf <VRF> ] ]

```

### Параметры

<ID> – номер OSPF-процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF для OSPF-процесса, в котором будет отображена информация по соседям, задается строкой до 31 символа.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

## Пример

```

esr# show ip ospf neighbors
Router ID      Pri      State      DTime      Interface  Router IP
160.0.0.2     0        full/ptp   00:53      vlink0     160.0.0.2
95.0.0.1      1        full/dr    00:31      gi1_15     115.0.0.10

```



```

10.100.100.2    128    full/ptp    00:37    gre_25    25.25.0.2
153.0.0.1      1      full/bdr   00:30    po1       1.1.0.2
10.100.100.2   128    2way/other 00:34    gi1_14.25 14.25.0.2
24.24.24.24    15     full/bdr   00:32    te1_1     24.0.0.2

```

## show ip ospf virtual-links

Данная команда отображает информацию о виртуальных соединениях.

### Синтаксис

```
show ip ospf <ID> [ vrf <VRF> ] virtual-links
```

### Параметры

<ID> – номер OSPF-процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF для OSPF-процесса, в котором будет отображена информация по виртуальным соединениям, задается строкой до 31 символа.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```

esr# show ip ospf 10 virtual-links
Virtual Link to router 160.0.0.2 is ptp
Peer IP: 160.0.0.2
Transit area: 1.1.1.1
Interface has no authentication
Timer intervals configured Hello 10, Dead 60, Retransmit 5, Wait 60
Adjacency State full

```

## show ipv6 ospf

Данная команда отображает таблицу маршрутизации OSPFv3, если не указан аргумент. При указании процесса выводит информацию о конфигурации интерфейсов по данному процессу.

### Синтаксис

```
show ip ospf <ID> [ vrf <VRF> ] database [ active ] [ area-id <AREA-ID> ] [ detailed ]
[ { asbr-summary | external | link-local | network | nssa-external | opaque-link | prefix
| router | summary } ] [ lsa-id <LSA-ID> ] [ router-id <ROUTER-ID> ] [ self-originate ]
[ statistic ]
```

### Параметры

active – показать LSA, время жизни которых менее 3600 секунд;

<AREA-ID> – идентификатор области, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

asbr-summary – показать LSA Type 0x2004 – Inter-Area-Router-LSA;

detailed – показать расширенную информацию о LSA;

external – показать LSA Type 0x4005 – AS-external-LSA;

link-local - показать LSA Type 0x0008 – Link-LSA;

<LSA-ID> – идентификатор LSA, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

network – показать LSA Type 0x2002 – Network-LSA;

nssa-external – показать LSA Type 0x2007 – NSSA-LSA;

opaque-link – показать Opaque LSA;

prefix – показать LSA Type 0x2009 – Intra-Area-Prefix-LSA;

router – показать LSA Type 0x2001 – Router-LSA;

<ROUTER-ID> – идентификатор маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

self-originate – показать локально сгенерированные LSA;

statistic – показать статистическую информацию о LSA, хранящихся в LSDB;

summary – показать LSA Type 0x2003 – Inter-Area-Prefix-LSA;

<ID> – номер OSPF-процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF для OSPF-процесса, в котором будет отображена таблица данных, задается строкой до 31 символа.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

Отображение таблицы маршрутизации:

```
esr# show ipv6 ospf
0          fc00::/120          [150/10] dev gi1/0/5          [ospf2 19:39:18] (2.2.2.2)
```

## show ipv6 ospf database

Данная команда отображает таблицу данных OSPFv3.

## Синтаксис

```
show ipv6 ospf <ID> [vrf <VRF>] database
```

## Параметры

<ID> – номер процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF для OSPFv3-процесса, в котором будет отображена таблица данных, задается строкой до 31 символа.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```

esr# show ipv6 ospf 111 database
Global
Type  LS ID           Router           Age  Sequence  Checksum
0005  10.166.11.12      10.1.0.1        1020 80000013   01b7
0005  0.0.0.0           10.166.11.1     245  80000010   aa48
0005  10.62.19.128     10.166.11.1     725  8000000e   6d2b
0005  10.62.20.0       10.166.11.1     731  8000000d   69af
0005  10.62.20.128    10.166.11.1     244  80000010   5e37
0005  10.62.21.128    10.166.11.1     244  80000010   5341
0005  10.166.11.0     10.166.11.1     245  80000010   cc6d
0005  10.166.11.12    10.166.11.1     245  80000010   54d9
Area 0.0.11.1
Type  LS ID           Router           Age  Sequence  Checksum
0001  10.1.0.1        10.1.0.1        1015 80000067   989e
0001  10.166.11.1    10.166.11.1     1021 80000018   8d96
0002  10.166.11.14   10.166.11.1     1021 80000001   68a5

```

## show ipv6 ospf interface

Данная команда отображает информацию об интерфейсах, на которых включен протокол OSPFv3.

## Синтаксис

```
show ipv6 ospf interface [ vrf <VRF> ] [ <IF> ]
```

## Параметры

<ID> – номер процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF для OSPFv3-процесса, в котором будет отображена информация об OSPFv3-интерфейсе, задается строкой до 31 символа;

<IF> – интерфейс или группы интерфейсов, задается в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

## Необходимый уровень привилегий

1

## Командный режим

ROOT

### Пример

```

esr# show ipv6 ospf interface gigabitethernet 1/0/14
esr# Interface:                gigabitethernet 1/0/14
Internet Address:              IID 0
Router ID:                     88.88.88.88
Network Type:                  broadcast
Area:                          0.0.0.0 (0)
Transmit:                      1
State:                         backup
Priority:                       128
Cost:                          10
ECMP weight:                   1
Hello timer:                   10
Wait timer:                    40
Dead timer:                    40
Retransmit timer:              5
Designed router (ID):          77.0.0.1
Designed router (IP):          fe80::c602:46ff:feed:0
Backup designed router (ID):  88.88.88.88
Backup designed router (IP):  fe80::1:2ff:fe03:463
Neighbor Count:                1
Adjacent neighbor count:       1
Adjacent with neighbor:        77.0.0.1 (dr)

```

## show ipv6 ospf neighbors

Данная команда отображает информацию о всех соседях или соседях определенного OSPFv3-процесса.

### Синтаксис

```
show ipv6 ospf [<ID> [vrf <VRF>]] neighbors
```

### Параметры

<ID> – номер процесса, принимает значения [1..65535], опциональный параметр.

<VRF> – имя экземпляра VRF для OSPFv3-процесса, в котором будет отображена информация по соседям, задается строкой до 31 символа.

### Необходимый уровень привилегий

1

## Командный режим

ROOT

### Пример

```

esr# show ipv6 ospf neighbor

```

Router ID	Pri	State	DTime	Interface	Router IP
77.0.0.1	1	full/dr	00:32	gi1_14	fe80::c602:46ff:feed:0
33.33.33.33	128	full/bdr	00:35	gi1_18	fe80::20:3ff:fea0:498

## show ipv6 ospf virtual-links

Данная команда отображает информацию о виртуальных соединениях.

### Синтаксис

```
show ipv6 ospf <ID> [vrf <VRF>] virtual-links
```

### Параметры

<ID> – номер процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF для OSPFv3-процесса, в котором будет отображена информация по виртуальным соединениям, задается строкой до 31 символа.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
esr# show ipv6 ospf 10 virtual-links
Virtual Link to router 160.0.0.2 is ptp
Peer IP: fe80::20:3ff:fea0:498
Transit area: 1.1.1.1
Interface has no authentication
Timer intervals configured: Hello 10, Dead 60, Retransmit 5, Wait 60
Adjacency State full
```

## summary-address

Командой включается суммаризация или скрытие подсетей. Суммаризация или скрытие подсетей при помощи данной команды возможны только для межзоновых маршрутов (типа IA).

Использование отрицательной формы команды (no) отключает суммаризацию или скрытие подсетей.

### Синтаксис

```
summary-address { <ADDR/LEN> | <IPV6-ADDR/LEN> } { advertise | not-advertise }
no summary-address <ADDR/LEN>
```

### Параметры

<ADDR/LEN> – IP-подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32];

<IPV6-ADDR/LEN> – IPv6-адрес и маска подсети, задаётся в виде X:X:X:X::X/EE, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128];

advertise – при указании команды вместо указанных подсетей будет анонсироваться суммарная подсеть;

not-advertise – при указании команды подсети, входящие в указанную подсеть, анонсироваться не будут.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-OSPF-AREA

CONFIG-OSPFV3-AREA

### Пример

```
esr(config-ospf-area)# summary-address 192.168.16.0/24
```

### virtual-link

Данной командой устанавливается виртуальное соединение между основной и удаленными областями, имеющими несколько областей между ними.

Использование отрицательной формы команды (no) удаляет созданное виртуальное соединение.

### Синтаксис

```
[no] virtual-link <ID>
```

### Параметры

<ID> – идентификатор маршрутизатора, с которым устанавливается виртуальное соединение, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-OSPF-AREA

CONFIG-OSPFV3-AREA

### Пример

```
esr(config-ospf-area)# virtual-link 160.0.0.2
```

### wait-interval

Данной командой определяется интервал времени в секундах, по истечении которого маршрутизатор выберет DR в сети.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

### Синтаксис

```
wait-interval <TIME>
```

```
no wait-interval
```

### Параметры

<TIME> – время в секундах, принимает значения [1..65535].

### Значение по умолчанию

40

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-OSPF-VLINK

CONFIG-OSPFV3-VLINK

### Пример

```
esr(config-ospf-vlink)# ospf wait-interval 60
```

## Настройка протокола IS-IS

- [address-family](#)
- [authentication area algorithm](#)
- [authentication area key](#)
- [authentication area key chain](#)
- [authentication domain algorithm](#)
- [authentication domain key](#)
- [authentication domain key chain](#)
- [clear isis](#)
- [hostname dynamic](#)
- [is-type](#)
- [isis adjacency point-to-point mode](#)
- [isis authentication algorithm](#)
- [isis authentication key](#)
- [isis authentication key chain](#)
- [isis circuit-type](#)
- [isis csnp-interval](#)
- [isis enable](#)
- [isis hello-interval](#)
- [isis hello-multiplier](#)
- [isis instance](#)
- [isis lsp-interval](#)
- [isis lsp-retransmit-interval](#)
- [isis metric](#)
- [isis network point-to-point](#)
- [isis priority](#)
- [isis psnp-interval](#)
- [lsp-refresh-interval](#)
- [max-lsp-lifetime](#)
- [metric-style](#)
- [net](#)
- [no isis hello-padding](#)
- [preference](#)
- [router isis](#)
- [show ip isis](#)
- [show ipv6 isis](#)
- [show isis database](#)
- [show isis hostname](#)
- [show isis neighbors](#)
- [show isis topology](#)
- [spf-timeout](#)

### address-family

Данная команда включает обмен маршрутами указанного семейства по протоколу IS-IS. Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

### Синтаксис

```
[no] address-family { ipv4 | ipv6 }
```



**Параметры**

- ipv4 – семейство адресов IPv4;
- ipv6 – семейство адресов IPv6.

**Значение по умолчанию**

ipv4 – включено;

ipv6 – отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-ISIS

**Пример**

```
esr(config-isis)# address-family ipv4
```

**authentication area algorithm**

Данная команда позволяет выбрать алгоритм аутентификации для L1-уровня, шифроваться будут LSP/SNP-пакеты.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

**Синтаксис**

```
authentication area algorithm <ALGORITHM>
no authentication area algorithm
```

**Параметры**

<ALGORITHM> – алгоритм аутентификации:

- cleartext – пароль, передается открытым текстом;
- md5 – пароль, хешируется по алгоритму md5.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-ISIS

## Пример

```
esr(config-isis)# authentication area algorithm cleartext
```

### authentication area key

Данная команда устанавливает пароль аутентификации для L1-уровня, этим паролем будут шифроваться LSP/SNP-пакеты.

Использование отрицательной формы команды (no) удаляет пароль.

### Синтаксис

```
authentication area key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no authentication area key
```

### Параметры

<CLEAR-TEXT> – пароль, задаётся строкой, длиной [8..16] символов;

<ENCRYPTED-TEXT> – зашифрованный пароль размером [8..16] байт ([16..32] символа) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-ISIS

## Пример

```
esr(config-isis)# authentication area key ascii-text password
```

### authentication area key chain

Данная команда устанавливает список ключей для аутентификации для L1-уровня.

Использование отрицательной формы команды (no) удаляет привязку к списку ключей.

### Синтаксис

```
authentication area key chain <KEYCHAIN>
no authentication area key chain
```

### Параметры

<KEYCHAIN> – идентификатор списка ключей, задаётся строкой [1..16] символов.

### Необходимый уровень привилегий

15

## Командный режим

CONFIG-ISIS

### Пример

```
esr(config-isis)# authentication area key chain key-1
```

### authentication domain algorithm

Данная команда устанавливает алгоритм аутентификации для L2-уровня, шифроваться будут LSP/SNP-пакеты.

Использование отрицательной формы команды (no) отключает аутентификацию.

### Синтаксис

```
authentication domain algorithm <ALGORITHM>  
no authentication domain algorithm
```

### Параметры

<ALGORITHM> – алгоритм аутентификации:

- cleartext – пароль, передается открытым текстом;
- md5 – пароль, хешируется по алгоритму md5.

### Необходимый уровень привилегий

15

## Командный режим

CONFIG-ISIS

### Пример

```
esr(config-isis)# authentication domain algorithm cleartext
```

### authentication domain key

Данная команда устанавливает пароль аутентификации для L2-уровня, этим паролем будут шифроваться LSP/SNP-пакеты.

Использование отрицательной формы команды (no) удаляет пароль.

### Синтаксис

```
authentication domain key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }  
no authentication domain key
```

## Параметры

<CLEAR-TEXT> – пароль, задаётся строкой, длиной [8..16] символов;

<ENCRYPTED-TEXT> – зашифрованный пароль размером [8..16] байт ([16..32] символа) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-ISIS

## Пример

```
esr(config-isis)# authentication domain key ascii-text password
```

## authentication domain key chain

Данная команда устанавливает список ключей для аутентификации для L2-уровня.

Использование отрицательной формы команды (no) удаляет привязку к списку ключей.

## Синтаксис

```
authentication domain key chain <KEYCHAIN>  
no authentication domain key chain
```

## Параметры

<KEYCHAIN> – идентификатор списка ключей, задаётся строкой [1..16] символов.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-ISIS

## Пример

```
esr(config-isis)# authentication domain key chain key-1
```

## clear isis

Данная команда перезапускает определенный IS-IS процесс. Команда без параметров перезапускает все процессы IS-IS, настроенные на маршрутизаторе.

### Синтаксис

```
clear isis [ <ID> ] [ vrf <VRF_NAME> ]
```

### Параметры

<ID> – номер процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример

```
esr# clear isis 15
```

### hostname dynamic

Данная команда включает использование TLV 137, которое позволяет передавать hostname маршрутизатора в LSP.

Использование отрицательной формы команды (no) отключает данную возможность.

### Синтаксис

```
[no] hostname dynamic
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-ISIS

### Пример

```
esr(config-isis)# hostname dynamic
```

**is-type**

Данная команда определяет, на каком уровне маршрутизации будет работать текущий процесс IS-IS. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
is-type <LEVEL>
no is-type
```

**Параметры**

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-1-2 – работа производится и на 1, и на 2 уровне;
- level-2-only – работа производится только на 2 уровне.

**Значение по умолчанию**

level-1-2

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-ISIS

**Пример**

```
esr(config-isis)# is-type level-1
```

**isis adjacency point-to-point mode**

Данная команда устанавливает способ установления соседства в режиме "точка-точка". Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
isis adjacency point-to-point mode <MODE>
no isis adjacency point-to-point mode
```

**Параметры**

<MODE> – режим установления соседства:

- two-way-only – ИИИ с TLV 240 "Point-to-Point Three-Way Adjacency" будут обрабатываться, но сам TLV 240 игнорируется;
- three-way-only – ИИИ без TLV 240 обрабатываться не будут;
- two-way-three-way – возможность работы как с TLV 240 так и без.

**Значение по умолчанию**

Разрешены оба способа.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

**Пример**

```
esr(config-if-gi)# isis adjacency point-to-point mode three-way-only
```

**isis authentication algorithm**

Данная команда устанавливает алгоритм аутентификации для hello-пакетов (IIH), выходящих с данного интерфейса.

Использование отрицательной формы команды (no) отключает аутентификацию.

**Синтаксис**

```
isis authentication algorithm <ALGORITHM> [ <LEVEL> ]
```

```
no isis authentication algorithm [ <LEVEL> ]
```

**Параметры**

<ALGORITHM> – алгоритм аутентификации:

- cleartext – пароль, передается открытым текстом;

- md5 – пароль, хешируется по алгоритму md5.

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

### Пример

```
esr(config-if-gi)# isis authentication algorithm cleartext level-1
```

### isis authentication key

Данная команда устанавливает пароль для аутентификации hello-пакетов (IIH).

Использование отрицательной формы команды (no) удаляет пароль.

### Синтаксис

```
isis authentication key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
[ <LEVEL> ]
no isis authentication key [ <LEVEL> ]
```

### Параметры

<CLEAR-TEXT> – пароль, задаётся строкой длиной [8..16] символов;



<ENCRYPTED-TEXT> – зашифрованный пароль размером [8..16] байт ([16..32] символа) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

### Пример

```
esr(config-if-gi)# isis authentication key ascii-text password level-1
```

### isis authentication key chain

Данная команда устанавливает список ключей для аутентификации hello-пакетов (IIH).

Использование отрицательной формы команды (no) удаляет привязку к списку ключей.

### Синтаксис

```
isis authentication key chain <KEYCHAIN> [ <LEVEL> ]
```

```
no isis authentication key chain [ <LEVEL> ]
```

### Параметры

<KEYCHAIN> – идентификатор списка ключей, задаётся строкой [1..16] символов.

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

### Пример

```
esr(config-if-gi)# isis authentication key chain key-1 level-1
```

### isis circuit-type

Данная команда определяет на каком уровне маршрутизации будет работать текущий процесс IS-IS на конкретном интерфейсе.

Использование отрицательной формы команды (no) устанавливает уровень работы протокола IS-IS на интерфейсе в соответствии с конфигурацией самого IS-IS процесса.

### Синтаксис

```
isis circuit-type <LEVEL>
no isis circuit-type
```

### Параметры

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-1-2 – работа производится и на 1, и на 2 уровне;
- level-2-only – работа производится только на 2 уровне.

## Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

### Пример

```
esr(config-if-gi)# isis circuit-type level-1
```

### isis csnp-interval

Данная команда устанавливает интервал генерации и отправки Complete Sequence Number Protocol Data Units (CSNP PDUs).

Использование отрицательной формы команды (no) устанавливает значению по умолчанию.

### Синтаксис

```
isis csnp-interval <TIME> [ <LEVEL> ]
no isis csnp-interval [ <LEVEL> ]
```

### Параметры

<TIME> – время в секундах, принимает значения [1..65535];

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

**Значение по умолчанию**

10

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

**Пример**

```
esr(config-if-gi)# isis csnp-interval 33 level-1
```

**isis enable**

Данная команда включает работу протокола IS-IS на интерфейсе.

Использование отрицательной формы команды (no) выключает IS-IS процесс.

**Синтаксис**

```
isis enable
```

```
no isis enable
```

**Параметры**

Команда не содержит параметров

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

**Пример**

```
esr(config-if-gi)# isis enable
```

**isis hello-interval**

Данная команда устанавливает интервал отправки hello-пакетов (ИП) с интерфейса маршрутизатора. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
isis hello-interval <TIME> [ <LEVEL> ]
no isis hello-interval [ <LEVEL> ]
```

**Параметры**

<TIME> – время в секундах, принимает значения [1..65535];

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;

- level-2 – работа производится только на 2 уровне.

### Значение по умолчанию

10

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

### Пример

```
esr(config-if-gi)# isis hello-interval 33 level-1
```

### isis hello-multiplier

Данная команда устанавливает множитель для вычисления и отправки Hold Time (Hold Time = hello-timer \* hello-multiplier) в hello пакетах (IIH).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
isis hello-multiplier <VALUE> [ <LEVEL> ]
no isis hello-multiplier [ <LEVEL> ]
```

### Параметры

<VALUE> – число, принимающее значения [3..1000];

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

### Значение по умолчанию

3

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

### Пример

```
esr(config-if-gi)# isis hello-multiplier 34 level-1
```

### isis instance

Данная команда определяет принадлежность интерфейса к определенному IS-IS процессу.

Использование отрицательной формы команды (no) удаляет принадлежность интерфейса к IS-IS процессу.

### Синтаксис

```
isis instance <ID>
```

```
no isis instance
```

## Параметры

<ID> – номер процесса, принимает значения [1..65535].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

## Пример

```
esr(config-if-gi)# isis instance 35
```

## isis lsp-interval

Данная команда задает интервал между успешными последовательными передачами Link-state Packets (LSPs) в Broadcast-сети.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
isis lsp-interval <TIME> [ <LEVEL> ]
```

```
no isis lsp-interval [ <LEVEL> ]
```

## Параметры

<TIME> – время в миллисекундах, принимает значения [1-10000];

<LEVEL> – уровень работы протокола IS-IS:



- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

### Значение по умолчанию

100

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

### Пример

```
esr(config-if-gi)# isis lsp-interval 36 level-2
```

### isis lsp-retransmit-interval

Данная команда задает интервал повторного распространения всех Link-state Packets (LSPs) в PtP-сети. Использование отрицательной формы команды (no) устанавливает значению по умолчанию.

### Синтаксис

```
isis lsp-retransmit-interval <TIME> [ <LEVEL> ]
no isis lsp-retransmit-interval [ <LEVEL> ]
```

### Параметры

<TIME> – время в секундах, принимает значения [1..65535];

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

### Значение по умолчанию

5

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

### Пример

```
esr(config-if-gi)# isis lsp-retransmit-interval 37 level-2
```

### isis metric

Данная команда устанавливает определенное значение метрики для интерфейса.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
isis metric <VALUE> [<LEVEL>]
```

```
no isis metric [<LEVEL>]
```

## Параметры

<VALUE> – число, принимающее значения [1..16777215];

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

## Значение по умолчанию

10

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

## Пример

```
esr(config-if-gi)# isis metric 38 level-1
```

## isis network point-to-point

Данная команда переводит текущий интерфейс в режим работы point-to-point протокола IS-IS.

Использование отрицательной формы команды (no) прекращает работу в данном режиме.

## Синтаксис

```
[no] isis network point-to-point
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Отключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

## Пример

```
esr(config-gre)# isis network point-to-point
```

## isis priority

Данная команда устанавливает приоритет при выборе DIS.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
isis priority <VALUE> [ <LEVEL> ]  
no isis priority [ <LEVEL> ]
```

## Параметры

<VALUE> – число, принимающее значения [0..127];

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

## Значение по умолчанию

64

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

## Пример

```
esr(config-if-gi)# isis priority 39 level-2
```

## isis psnp-interval

Данная команда устанавливает интервал генерации и отправки Partial Sequence Number Protocol Data Units (PSNP PDUs).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
isis psnp-interval <TIME> [ <LEVEL> ]  
no isis psnp-interval [ <LEVEL> ]
```

### Параметры

<TIME> – время в секундах, принимает значения [1..65535];

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

### Значение по умолчанию

2

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

## Пример

```
esr(config-if-gi)# isis psnp-interval 40 level-1
```

## lsp-refresh-interval

Данная команда устанавливает интервалы обновления/генерации собственных LSP.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
lsp-refresh-interval { min | max } <TIME> [ <LEVEL> ]
no lsp-refresh-interval { min | max } [ <LEVEL> ]
```

### Параметры

min – минимальный интервал обновления/генерации;

max – максимальный интервал обновления/генерации;

<TIME> – время в секундах, принимает значения [1..65535];

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

### Значение по умолчанию

min 30

max 900

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-ISIS

## Пример

```
esr(config-isis)# lsp-refresh-interval min 41 level-1
```

## max-lsp-lifetime

Данная команда задает время жизни собственных LSP.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
max-lsp-lifetime <TIME> [ <LEVEL> ]
no max-lsp-lifetime [ <LEVEL> ]
```

## Параметры

<TIME> – время в секундах, принимает значения [1..65535];

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

## Значение по умолчанию

1200

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ISIS

## Пример

```
esr(config-isis)# max-lsp-lifetime 42 level-2
```

## metric-style

Данная команда устанавливает тип метрики, который будет использоваться в работе IS-IS процесса. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
metric-style { narrow | wide | transition } [ <LEVEL> ]
no metric-style [ <LEVEL> ]
```

## Параметры

narrow – информация о достижимости сетей распространяется с помощью TLV 128;

wide – информация о достижимости сетей распространяется с помощью TLV 135;

transition – информация о достижимости сетей распространяется обоими способами;

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

## Значение по умолчанию

narrow

## Необходимый уровень привилегий

10



## Командный режим

CONFIG-ISIS

### Пример

```
esr(config-isis)# metric-style wide level-1
```

## net

Данная команда устанавливает NET (Network Entity Title) адрес. Он репрезентует AFI (Authority and Format Identifier), Area ID (зону, к которой принадлежит маршрутизатор), System ID (идентификатор маршрутизатора) и значение Selector.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
[no] net <NET>
```

### Параметры

<NET> – NET-адрес, формат: ff[.ffff.ffff.ffff.ffff.ffff].ffff.ffff.ffff.00, где f – может принимать значения от [1..F].

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-ISIS

### Пример

```
esr(config-isis)# net 49.0123.1010.1010.1010.00
```

## no isis hello-padding

Данная команда задает отключает режим использования TLV 8 в hello-пакетах (IIH).

Использование положительной формы команды (без ключа no) устанавливает значение по умолчанию.

### Синтаксис

```
[no] isis hello-padding
```

### Параметры

Команда не содержит параметров.

**Значение по умолчанию**

Включено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

**Пример**

```
esr(config-if-gi)# isis hello-padding
```

**preference**

Данная команда устанавливает приоритетность маршрутов данного IS-IS процесса для основной таблицы маршрутизации.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
preference <VALUE>
```

```
no preference
```

**Параметры**

<VALUE> – принимает значения [1..255].

**Значение по умолчанию**

160

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-ISIS

**Пример**

```
esr(config-isis)# preference 43
```

**router isis**

Данная команда создаёт IS-IS процесс и осуществляет переход в режим настройки параметров этого процесса.

Использование отрицательной формы команды (no) удаляет IS-IS процесс из системы.

**Синтаксис**

```
[no] router isis <ID> [ vrf <VRF> ]
```

**Параметры**

<ID> – номер процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# router isis 44
```

**show ip isis**

Данная команда выводит внутреннюю таблицу маршрутизации.

**Синтаксис**

```
show ip isis [ <ID> ] [ <ADDR/LEN> ] [ <LEVEL> ] [ vrf <VRF> ]
```

## Параметры

<ID> – номер процесса, принимает значения [1..65535];

<ADDR/LEN> – адрес, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32];

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
esr# show ip isis 45
IPv4 local Level 1 RIB for IS-IS process
Last update: 2020-01-24 06:51:02.110
Address                Type Metric Via
3.3.3.11/32            L1    20    2.2.2.11 on gi1/0/2

IPv4 local Level 2 RIB for IS-IS process
Last update: 2020-01-24 06:51:12.440
Address                Type Metric Via
3.3.3.22/32            IA    20    1.1.1.2 on gi1/0/1
```

## show ipv6 isis

Данная команда выводит внутреннюю ipv6-таблицу маршрутизации.

## Синтаксис

```
show ipv6 isis [ <ID> ] [ <IPV6-ADDR/LEN> ] [ <LEVEL> ] [ vrf <VRF> ]
```

## Параметры

<ID> – номер процесса, принимает значения [1..65535];

<IPV6-ADDR/LEN> – IPv6-адрес подсети с маской, задаётся в виде X:X:X::X/EE, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128];

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример**

```

esr# show ipv6 isis 46
IPv6 local Level 1 RIB for IS-IS process
Last update: 2020-01-24 07:08:37.761
Address                Type Metric Via
6666::/64              L1    20    fe80::aaf9:4bff:feab:813a on gi1/0/2

IPv6 local Level 2 RIB for IS-IS process
Last update: 2020-01-24 07:07:22.917
Address                Type Metric Via
2001::/64              IA    20    fe80::aaf9:4bff:feac:b4a1 on gi1/0/1

```

**show isis database**

Данная команда выводит информацию по локальной базе всех полученных и сгенерированных LSP (LSDB).

**Синтаксис**

```
show isis database [ <ID> ] [ <LEVEL> ] [ <LSP-ID> ] [ detailed ] [ vrf <VRF> ]
```

**Параметры**

<ID> – номер процесса, принимает значения [1..65535];

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

<LSP-ID> – ID LSP пакета, задаётся в виде xxxx.xxxx.xxxx.yy-zz, где на месте x – System-ID, y – Pseudonode-ID, z – номер фрагмента. Все числа могут принимать значения [0..F];

detailed – развёрнутая информация о выбранном пункте;

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

## Пример

```

esr# show isis database 47
IS-IS Level 1 Link State Database
LSPID                Hostname          LSP Seq Num  LSP Checksum  LSP Holdtime  ATT/P/OL
1010.1010.1010.00-00 * esr-l1l2        0x0000000f   0x5eba        960            1/0/0
1111.1111.1111.00-00 esr-l1            0x00000010   0x4402        1071           0/0/0
1111.1111.1111.01-00 esr-l1            0x00000004   0x1a99        985            0/0/0

IS-IS Level 2 Link State Database
LSPID                Hostname          LSP Seq Num  LSP Checksum  LSP Holdtime  ATT/P/OL
1010.1010.1010.00-00 * esr-l1l2        0x00000010   0x7776        1082           0/0/0
1222.1222.1222.01-00 esr-l2            0x00000003   0xb87a        997            0/0/0
1222.1222.1222.00-00 esr-l2            0x0000000e   0x84c4        997            0/0/0

```

## show isis hostname

Данная команда выводит соответствие System-id и имени роутера.

## Синтаксис

```
show isis hostname [ <ID> ] [ vrf <VRF> ]
```

## Параметры

<ID> – номер процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```

esr# show isis hostname 48
IS-IS 12
Level System ID          Dynamic Hostname
  1   1010.1010.1010 *   esr-l1l2
  1   1111.1111.1111     esr-l1
  2   1010.1010.1010 *   esr-l1l2
  2   1222.1222.1222     esr-l2

```

## show isis neighbors

Данная команда отображает информацию по состоянию IS-IS соседств.

## Синтаксис

```
show isis neighbors [ <ID> ] [ <LEVEL> ] [ detailed ] [ vrf <VRF> ]
```

## Параметры

<ID> – номер процесса, принимает значения [1..65535];

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

detailed – развёрнутая информация о выбранном пункте;

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```

esr# show isis neighbors 49 detailed
IS-IS Level 1 Neighbors
System ID      Hostname      Interface      State      Holdtime  SNPA
1111.1111.1111 esr-l1        gi1/0/2        Up         10        a8f9.4bab.813a

IS-IS Level 2 Neighbors
System ID      Hostname      Interface      State      Holdtime  SNPA
1222.1222.1222 esr-l2        gi1/0/1        Up         10        a8f9.4bac.b4a1

```

## show isis topology

Данная команда выводит полную таблицу достижимости всех сетей и узлов.

## Синтаксис

```
show isis topology [ <ID> ] [ <LEVEL> ] [ ipv4 | ipv6 ] [ vrf <VRF> ]
```

## Параметры

<ID> – номер процесса, принимает значения [1..65535];

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне;

ipv4 – семейство адресов IPv4;

ipv6 – семейство адресов IPv6;

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

### Пример

```

esr# show isis topology 50 level-2 Ipv4
IS-IS path to Level 2 routers (narrow metric-style)
Last update: 2020-01-24 07:08:47.783
Vertex                Type                Metric  Next-Hop           Interface
1.1.1.0/24            IPv4 Internal        0       --                 --

IS-IS path to Level 2 routers (wide metric-style)
Last update: 2020-01-24 07:08:47.783
Vertex                Type                Metric  Next-Hop           Interface
3.3.3.22/32          IPv4 IA              20      esr-l2             gi1/0/1
1.1.1.0/24            IPv4                  0       --                 --

```

## spf-timeout

Данная команда устанавливает таймаут перед следующим расчётом SPF.

Использование отрицательной формы команды (no) устанавливает значению по умолчанию.

### Синтаксис

```

spf-timeout <TIME> [ <LEVEL> ]
no spf-timeout [ <LEVEL> ]

```

### Параметры

<TIME> – время в миллисекундах, принимает значения [1..10000];

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

### Значение по умолчанию

5000

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-ISIS

### Пример

```

esr(config-isis)# spf-timeout 51 level-1

```



## 24 Настройки MPLS

- Общие настройки MPLS
  - forwarding interface
  - history statistics
  - l2vpn
  - ldp
  - mpls
  - system cpu load-balance mpls passenger
- Настройки протокола LDP
  - address-family
  - advertise-labels
  - clear mpls ldp
  - description
  - discovery hello holdtime
  - discovery hello interval
  - discovery targeted-hello accept
  - discovery targeted-hello interval
  - discovery targeted-hello holdtime
  - egress-label-type
  - enable
  - interface
  - keepalive
  - neighbor
  - password
  - prefix-list
  - router-id
  - show mpls ldp binding
  - show mpls ldp discovery
  - show mpls forwarding-table
  - show mpls ldp neighbor
  - targeted
  - transport-address
- Настройки MPLS L2VPN
  - autodiscovery bgp
  - ignore encapsulation-mismatch
  - ignore mtu-mismatch
  - mtu
  - rd
  - route-target
  - ve id
  - ve range
  - vpn id
  - bridge-group
  - description
  - enable
  - encapsulation mpls mtu
  - encapsulation mpls status-tlv disable
  - interface
  - neighbor-address
  - p2p
  - pw
  - pw-class
  - pw-class
  - transport-mode
  - vpls
  - show mpls l2vpn bindings

- [show mpls l2vpn pseudowire](#)
- [show mpls l2vpn p2p](#)
- [show mpls l2vpn pw-class](#)
- [show mpls l2vpn vpls](#)

## Общие настройки MPLS

### forwarding interface

Данной командой включается передача MPLS-пакетов на интерфейсе.

Использование отрицательной формы команды (no) отключает передачу MPLS-пакетов на интерфейсе.

### Синтаксис

```
[no] forwarding interface { <IF> | <TUN> }
```

### Параметры

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#).

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-MPLS

### Пример

```
esr(config-mpls)# forwarding interface gi 1/0/14
```

### history statistics

Данной командой включается запись статистики использования псевдопровода/динамического псевдопровода.

Использование отрицательной формы команды (no) отключает запись статистики использования псевдопровода/динамического псевдопровода.

### Синтаксис

```
[no] history statistics
```

**Параметры**

Отсутствуют.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-L2VPN-AUTODISCOVERY-BGP

CONFIG-L2VPN-PW

**Пример**

```
esr(config-l2vpn-autodiscovery-bgp)# history statistics
```

**l2vpn**

Данной командой осуществляется переход в контекст настройки MPLS-L2-туннелей.

Использование отрицательной формы команды (no) удаляет все настройки MPLS-L2-туннели.

**Синтаксис**

[no] l2vpn

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-MPLS

**Пример**

```
esr(config-mpls)# l2vpn
```

**ldp**

Данной командой осуществляется переход в контекст настройки протокола LDP.

Использование отрицательной формы команды (no) удаляет все настройки протокола LDP.

**Синтаксис**

[no] ldp

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-MPLS

## Пример

```
esr(config-mpls)# ldp
```

## mpls

Данной командой осуществляется переход в контекст настройки функционала MPLS.

Использование отрицательной формы команды (no) удаляет все настройки функционала LDP.

## Синтаксис

[no] mpls

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# mpls
```

## system cpu load-balance mpls passenger

Данной командой включается функционал балансировки трафика между CPU маршрутизатора.

Использование отрицательной формы команды (no) отключает функционал балансировки.

## Синтаксис

system cpu load-balance mpls passenger [ <OPTION> ]

no system cpu load-balance mpls passenger [ <OPTION> ]

## Параметры

Параметр <OPTION> принимает значения:

- ip – включает поиск и добавление в расчет хеша ip-src и ip-dst;
- ipoe-pw-with-cw – возможность явно указать, что используется заголовок с Control Word.
- ipoe-pw-without-cw – возможность явно указать, что не используется заголовок с Control Word.

## Значение по умолчанию

Отключено. В расчете хеш не используются ip-src и ip-dst.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# system cpu load-balance mpls passenger ip
```

## Настройки протокола LDP

### address-family

Команда позволяет перейти в режим конфигурации параметров LDP для выбранной address family. Использование отрицательной формы команды (no) удаляет все параметры из конфигурации.

### Синтаксис

```
[no] address-family ipv4
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-LDP

### Пример

```
esr(config-ldp)# address-family ipv4
```

## advertise-labels

Команда позволяет задать FEC, для которых протокол LDP будет выделять и распространять метки. Использование отрицательной формы команды (no) возвращает значение по умолчанию.

### Синтаксис

```
advertise-labels <OBJ-GROUP-NETWORK-NAME>  
no advertise-labels
```

### Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа.

### Значение по умолчанию

Метки выделяются и анонсируются для всех FEC.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-LDP

### Пример

```
esr(config-ldp)# advertise-labels mpls_adv
```

## clear mpls ldp

Данная команда служит для сброса LDP-процесса или для очистки таблицы LDP-соседей.

### Синтаксис

```
clear mpls ldp [neighbor <ADDR> ]
```

### Параметры

<ADDR> – IP-адрес LDP-соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

### Необходимый уровень привилегий

1

### Командный режим

ROOT

**Пример**

```
ESR1# clear mpls ldp neighbor 1.1.1.1
```

**description**

Данная команда позволяет добавить описание для LDP-соседа.

Использование отрицательной формы команды (no) удаляет описание LDP-соседа.

**Синтаксис**

```
description <STRING>
no description
```

**Параметры**

<STRING> – строка от 1 до 255 символов.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-LDP-NEIGH

**Пример**

```
esr(config-ldp-neig)# description ESR1
```

**discovery hello holdtime**

Данная команда устанавливает значение параметра holdtime. Holdtime – интервал времени, в течении которого должно прийти хотя бы одно LDP hello сообщение.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
discovery hello holdtime <TIME>
no discovery hello holdtime
```

**Параметры**

<TIME> – время в секундах в интервале [3..65535].

**Значение по умолчанию**

15

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-LDP

CONFIG-LDP-AF-IPV4-IF

**Пример**

```
esr(config-ldp-af-ipv4-if)# discovery hello holdtime 45
```

**discovery hello interval**

Данная команда устанавливает значение параметра hello interval. Hello interval – промежуток времени между отправкой LDP hello сообщения.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
discovery hello interval <TIME>
```

```
no discovery hello interval
```

**Параметры**

<TIME> – время в секундах в интервале [1..65535].

**Значение по умолчанию**

5

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-LDP

CONFIG-LDP-AF-IPV4-IF

**Пример**

```
esr(config-ldp-af-ipv4-if)# discovery hello interval 15
```

**discovery targeted-hello accept**

Данная команда разрешает прием targeted-hello сообщений.

Использование отрицательной формы команды (no) запрещает прием targeted-hello сообщений.



**Синтаксис**

```
discovery targeted-hello accept  
no targeted-hello accept
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-LDP-AF-IPV4

**Пример**

```
esr(config-ldp-af-ipv4)# discovery targeted-hello accept
```

**discovery targeted-hello interval**

Данная команда устанавливает значение параметра hello interval. Hello interval – промежуток времени между отправкой LDP hello сообщений для targeted-соседа.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
discovery targeted-hello interval <TIME>  
no discovery targeted-hello interval
```

**Параметры**

<TIME> – время в секундах в интервале [1..65535].

**Значение по умолчанию**

5

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-LDP

CONFIG-LDP-NEIGH

## Пример

```
esr(config-ldp-neig)# discovery targeted-hello interval 15
```

### discovery targeted-hello holdtime

Данная команда устанавливает значение параметра holdtime для targeted-соседа. Holdtime – интервал времени, в течение которого должно прийти хотя бы одно LDP hello сообщение.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
discovery targeted-hello holdtime <TIME>
```

```
no discovery targeted-hello holdtime
```

### Параметры

<TIME> – время в секундах в интервале [3..65535].

### Значение по умолчанию

45

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-LDP

CONFIG-LDP-NEIGH

## Пример

```
esr(config-ldp-af-ipv4-if)# discovery hello holdtime 45
```

### egress-label-type

С помощью данной команды устанавливается режим, когда для FEC, у которых маршрутизатор является egress lsr, анонсируется метка explicit-null (0) вместо implicit null (3).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
egress-label-type explicit-null
```

```
no egress-label-type
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Анонсируется IMPLICIT NULL (3).

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-LDP

**Пример**

```
esr(config-ldp)# egress-label-type explicit-null
```

**enable**

Данной командой включает процесс LDP.

Использование отрицательной формы команды (no) отключает процесс LDP.

**Синтаксис**

[no] enable

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-LDP

**Пример**

```
esr(config-ldp)# enable
```

## interface

Данная команда включает прием и рассылку LDP hello сообщений на интерфейсе для определенной address-family.

Использование отрицательной формы команды (no) отключает прием и рассылку LDP hello сообщений на интерфейсе.

### Синтаксис

```
[no] interface { <IF> | <TUN> }
```

### Параметры

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#).

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-LDP-AF-IPV4

### Пример

```
esr(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1
```

## keepalive

С помощью данной команды устанавливается время ожидания от соседа keepalive-сообщения. В случае истечения таймера сосед считается недоступным.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
keepalive <TIMER>
```

```
no keepalive
```

### Параметры

<TIMER> – время в секундах [3..65535].

### Значение по умолчанию

180

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-LDP

CONFIG-LDP-NEIGH

**Пример**

```
esr(config-ldp)# keepalive 120
esr(config-ldp-neig)# keepalive 160
```

**neighbor**

Данная команда позволяет создать и перейти в режим конфигурации LDP-targeted соседа. Использование отрицательной формы команды (no) удаляет targeted-соседа из конфигурации.

**Синтаксис**

```
[no] neighbor <ADDR>
```

**Параметры**

<ADDR> – IP-адрес соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-LDP

**Пример**

```
esr(config-ldp)# neighbor 2.2.2.2
```

**password**

Данная команда устанавливает пароль для аутентификации с соседом. Использование отрицательной формы команды (no) удаляет пароль.

**Синтаксис**

```
password { <TEXT> | encrypted <ENCRYPTED-TEXT> }
no password
```

## Параметры

<CLEAR-TEXT> – пароль, задаётся строкой длиной [8..16] символов;

<ENCRYPTED-TEXT> – зашифрованный пароль размером [8..16] байт ([16..32] символа) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-LDP-NEIGH

## Пример

```
esr(config-ldp-neig)# password cleartextpassword
```

## prefix-list

Данная команда устанавливает список, описывающий подсети, для которых протокол LDP будет выделять и распространять метки.

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

## Синтаксис

```
prefix-list <PREFIX-LIST-NAME>
```

```
no prefix-list
```

## Параметры

<PREFIX-LIST-NAME> – имя списка, задаётся строкой до 31 символа.

## Значение по умолчанию

Метки выделяются и анонсируются для всех FEC.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-LDP-AF-IPV4

## Пример

```
esr(config-ldp-af-ipv4)# prefix-list LDP_ALLOCATE
```

## router-id

Данной командой устанавливается идентификатор маршрутизатора для LDP-протокола.

Использование отрицательной формы команды (no) удаляет идентификатор маршрутизатора для LDP-протокола.

### Синтаксис

```
router-id { <ID> | <IF> | <TUN> }
no router-id
```

### Параметры

<ID> – идентификатор маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IF> – интерфейс, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#).

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-LDP

### Пример

```
esr(config-ldp)# router-id 1.1.1.1
```

## show mpls ldp binding

Данная команда позволяет вывести информацию по выделенным меткам.

### Синтаксис

```
show mpls ldp binding [ detailed ] [ { <ADDR/LEN> | neighbor <ADDR> | [ local <LABEL-ID> ] [ remote <LABEL-ID> ] } ]
```

### Параметры

<ADDR/LEN> – IP-подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32];

<LABEL-ID> – метка, задается в виде числа в диапазоне [0-104875];

<ADDR> – IP-адрес LDP-соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

detailed – информация выдается в детализированном формате.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```

ESR1# sh mpls ldp bindings
10.10.0.4/32:
  local label:    16
  remote label:  16      lsr: 10.10.0.1
  remote label: 149      lsr: 10.10.0.3
  remote label: imp-null lsr: 10.10.0.4      inuse
10.10.0.2/32:
  local label:    imp-null
  remote label:  34      lsr: 10.10.0.1
  remote label: 152      lsr: 10.10.0.3
  remote label: 142      lsr: 10.10.0.4
10.10.0.1/32:
  local label:    20
  remote label:  imp-null lsr: 10.10.0.1      inuse
  remote label: 139      lsr: 10.10.0.3
  remote label: 127      lsr: 10.10.0.4

```

## show mpls ldp discovery

Данная команда позволяет вывести оперативную информацию по состоянию процесса обнаружения LDP.

## Синтаксис

```
show mpls ldp discovery [ detailed ]
```

## Параметры

detailed – информация выдается в детализированном формате.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```

ESR1# show mpls ldp discovery detailed
Local LDP ID: 1.1.1.1
Discovery sources:
Interfaces:

```



```

gigabitethernet 1/0/1:
Hello interval: 5 seconds
Transport IP address: 1.1.1.1
LDP ID: 4.4.4.4
Source IP address: 10.10.10.2
Transport IP address: 4.4.4.4
Hold time: 15 seconds
Proposed hold time: 15/15 (local/peer) seconds
Targeted hellos:
1.1.1.1 -> 4.4.4.4:
Hello interval: 5 seconds
Transport IP address: 1.1.1.1
LDP ID: 4.4.4.4
Source IP address: 4.4.4.4
Transport IP address: 4.4.4.4
Hold time: 45 seconds
Proposed hold time: 45/45 (local/peer) seconds

```

## show mpls forwarding-table

Данная команда позволяет вывести таблицу коммутации по меткам (LFIB).

### Синтаксис

```
show mpls forwarding-table [ { <ADDR/LEN> | label <LABEL-ID> | nexthop <ADDR> | tunnel
<ID> } ]
```

### Параметры

<ADDR/LEN> – IP-подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32];

<LABEL-ID> – метка, задаётся в виде числа в диапазоне [0..104875];

<ADDR> – IP-адрес LDP-соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<ID> – идентификатор в виде числа в диапазоне [1..2000000].

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```

ESR1# show mpls forwarding-table
Local      Outgoing          Prefix              Outgoing
Next Hop
label      label              or tunnel ID Interface
-----
18         16                 PW   ID 100             --              4.4.4
.4

```

```
17      imp-null      4.4.4.4/32      gi1/0/1      10.10.1  
0.2
```

## show mpls ldp neighbor

Данная команда отображает информацию обо всех LDP-соседях.

### Синтаксис

```
show mpls ldp neighbor [ { <ADDR> | interface { <IF> | <TUN> } } ]
```

### Параметры

<ADDR> – IP-адрес LDP-соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#).

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
ESR1# show mpls ldp neighbor  
Peer LDP ID: 4.4.4.4; Local LDP ID 1.1.1.1  
State: Operational  
TCP connection: 4.4.4.4:38759 - 1.1.1.1:646  
Messages sent/received: 195/194  
Uptime: 03:07:33  
LDP discovery sources:  
gigabitethernet 1/0/1  
1.1.1.1 -> 4.4.4.4
```

## targeted

Данная команда включает отправку targeted hello сообщений.

Использование отрицательной формы команды (no) отключает отправку targeted hello сообщений.

### Синтаксис

```
[no] targeted
```

### Параметры

Команда не имеет параметров.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-LDP-NEIGH

### Пример

```
esr(config-ldp-neig)# targeted
```

### transport-address

С помощью данной команды определяется IP-адрес источника для LDP-сообщений, необходимых для установления сессии.

Использование отрицательной формы команды (no) удаляет сконфигурированный IP-адрес источника для LDP-сообщений.

### Синтаксис

```
transport-address <ADDR>  
no transport-address
```

### Параметры

<ADDR> – задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-LDP-AF-IPV4

### Пример

```
esr(config-ldp-af-ipv4)# transport-address 1.1.1.1
```

## Настройки MPLS L2VPN

### autodiscovery bgp

Команда позволяет перейти в контекст настройки BGP auto-discovery and signaling.

Использование отрицательной формы команды (no) удаляет параметр из конфигурации.

**Синтаксис**

```
[no] autodiscovery bgp
```

**Параметры**

Отсутствуют.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-L2VPN-EOMPLS

**Пример**

```
esr(config-l2vpn-vpls)# autodiscovery bgp
```

**ignore encapsulation-mismatch**

Данная команда разрешает установку псевдопроводного соединения при несовпадении типа инкапсуляции.

Использование отрицательной формы команды (no) включает проверку типа инкапсуляции перед установкой псевдопроводного соединения.

**Синтаксис**

```
[ no ] ignore encapsulation-mismatch
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-L2VPN-AUTODISCOVERY-BGP

**Пример**

```
esr(config-l2vpn-autodiscovery-bgp)# ignore encapsulation-mismatch
```

## ignore mtu-mismatch

Данная команда разрешает установку псевдопроводного соединения при несовпадении значения MTU. Использование отрицательной формы команды (no) включает проверку значения MTU перед псевдопроводного соединения.

### Синтаксис

```
[no] ignore mtu-mismatch
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-L2VPN-AUTODISCOVERY-BGP

### Пример

```
esr(config-l2vpn-autodiscovery-bgp)# ignore mtu-mismatch
```

## mtu

Данная команда задает значение MTU для псевдопроводного соединения. Заданное значение передается в extended community.

Использование отрицательной формы команды (no) устанавливает значение MTU по умолчанию.

### Синтаксис

```
mtu <VALUE>
```

```
no mtu
```

### Параметры

<VALUE> – размер MTU, принимает значения в диапазоне [552..10000].

### Значение по умолчанию

1500

### Необходимый уровень привилегий

10

**Командный режим**

CONFIG-L2VPN-AUTODISCOVERY-BGP

**Пример**

```
esr(config-l2vpn-autodiscovery-bgp)# mtu 1510
```

**rd**

Установка значения Route Distinguisher для данного экземпляра VPLS.

Использование отрицательной формы команды (no) удаляет параметр из конфигурации.

**Синтаксис**

```
rd <RD>
```

```
no rd
```

**Параметры**

<RD> – значение Route distinguisher, задается в одном из следующем виде:

- <ASN>:<nn>, где <ASN> – принимает значение [1..65535], nn – принимает значение [1..65535];
- <ADDR>:<nn>, где <ADDR> имеет вид – AAA.BBB.CCC.DDD/EE и AAA-DDD принимают значения [0..255], а nn – принимает значение [1..65535];
- <4BASN>:<nn>, где <4ASN> – принимает значение [1..4294967295], nn – принимает значение [1..65535].

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-L2VPN-AUTODISCOVERY-BGP

**Пример**

```
esr(config-l2vpn-autodiscovery-bgp)# rd 65000:10001
```

**route-target**

Установка перечня значений route-target extended community.

**Синтаксис**

```
[no] route-target { import | export | both } <RT>
```

## Параметры

`import` – команда определяет маршруты из таблицы L2VPN со значением <RT>, которые будут помещены в соответствующий экземпляр VPLS;

`export` – команда устанавливает значения <RT>, с которыми маршруты из данного экземпляра VPLS будут анонсированы в BGP;

`both` – команда устанавливает значение <RT>, с которым маршруты будут анонсированы и импортированы в данном экземпляре VPLS;

<RT> – значение `route-target`, задается в одном из следующем виде:

- <ASN>:<nn>, где <ASN> – принимает значение [1..65535], nn – принимает значение [1..65535];
- <ADDR>:<nn>, где <ADDR> имеет вид – AAA.BBB.CCC.DDD/EE и AAA-DDD принимают значения [0..255], а nn – принимает значение [1..65535];
- <4ASN>:<nn>, где <4ASN> – принимает значение [1..4294967295], nn – принимает значение [1..65535].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-L2VPN-AUTODISCOVERY-BGP

## Пример

```
esr(config-l2vpn-autodiscovery-bgp)# route-target export 65000:10001
```

## ve id

Данной командой задается идентификатор экземпляра VPLS.

Использование отрицательной формы команды (`no`) удаляет параметр из конфигурации.

## Синтаксис

```
ve id <ID>
```

```
no ve id
```

## Параметры

<ID> – идентификатор экземпляра VPLS, принимает значения в диапазоне [1..16384].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-L2VPN-AUTODISCOVERY-BGP

## Пример

```
esr(config-l2vpn-autodiscovery-bgp)# ve id 5
```

## ve range

Данной командой настраивается диапазон идентификаторов пограничных устройств VPLS.

Использование отрицательной формы команды (no) устанавливает значение ve range по умолчанию.

## Синтаксис

```
ve range <ID>
```

```
no ve range
```

## Параметры

<ID> – идентификатор пограничного устройства, принимает значения в диапазоне [8..100].

## Необходимый уровень привилегий

10

## Значение по умолчанию

10

## Командный режим

CONFIG-L2VPN-AUTODISCOVERY-BGP

## Пример

```
esr(config-l2vpn-autodiscovery-bgp)# ve range 20
```

## vpn id

Данной командой задается идентификатор VPN-соединения.

Использование отрицательной формы команды (no) удаляет параметр из конфигурации.

## Синтаксис

```
vpn id <ID>
```

```
no vpn id
```

## Параметры

<ID> – идентификатор VPN-соединения, принимает значения в диапазоне [1..4294967295].



**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-L2VPN-AUTODISCOVERY-BGP

**Пример**

```
esr(config-l2vpn-autodiscovery-bgp)# vpn id 5
```

**bridge-group**

С помощью данной команды задается VSI (Virtual Switching Instance).

Использование отрицательной формы команды (no) удаляет параметр из конфигурации.

**Синтаксис**

```
bridge-group <INDEX>
```

```
no bridge-group
```

**Параметры**

<INDEX> – индекс bridge-group. Bridge с указанным индексом должен быть заранее добавлен в конфигурацию.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-L2VPN-EOMPLS

**Пример**

```
esr(config-l2vpn-vpls)# bridge-group 25
```

**description**

Данной командой задается описание.

Использование отрицательной формы команды (no) удаляет описание.

**Синтаксис**

```
description <LINE>
```

```
no description
```

## Параметры

<LINE> – описание. Задается в виде строки длиной [1-255] символов.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-L2VPN-PW

CONFIG-L2VPN-PW-CLASS

## Пример

```
esr(config-l2vpn-pw)# description PW_FOR_VLAN398
```

## enable

Данной командой активируется выбранный функционал.

Использование отрицательной формы команды (no) деактивирует выбранный функционал.

## Синтаксис

[no] enable

## Параметры

Команда не имеет параметров.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-L2VPN-PW

CONFIG-L2VPN-EOMPLS

## Пример

```
esr(config-l2vpn-pw)# enable
```

## encapsulation mpls mtu

Данной командой устанавливается значение MTU для pseudowire-интерфейсов входящих в данный pw-class.

Использование отрицательной формы команды (no) устанавливает значение MTU по умолчанию.

### Синтаксис

```
encapsulation mpls mtu <MTU>  
no encapsulation mpls mtu
```

### Параметры

<MTU> – значение MTU, принимает значение в диапазоне [552..10000].

### Значение по умолчанию

1500

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-L2VPN-PW-CLASS

### Пример

```
esr(config-l2vpn-pw-class)# encapsulation mpls mtu 1464
```

### encapsulation mpls status-tlv disable

Данной командой отключается обмен status-tlv сообщениями.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
[no] encapsulation mpls status-tlv disable
```

### Параметры

Команда не имеет параметров.

### Значение по умолчанию

status-tlv enable

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-L2VPN-PW-CLASS

## Пример

```
esr(config-l2vpn-pw-class)# encapsulation mpls status-tlv disable
```

## interface

С помощью данной команды задается Attached Circuit интерфейс.

Использование отрицательной формы команды (no) удаляет Attached Circuit интерфейс.

## Синтаксис

```
interface { <IF> | <TUN> }
no interface
```

## Параметры

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#).

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-L2VPN-EOMPLS

## Пример

```
esr(config-l2vpn-p2p)# interface gigabitethernet 1/0/4.398
```

## neighbor-address

С помощью данной команды указывается адрес маршрутизатора, до которого будет установлен псевдопровод.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
neighbor-address <ADDR>
no neighbor-address
```

## Параметры

<ADDR> – IP-адрес маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

**Значение по умолчанию**

Адрес соседа принимается равным LSR\_ID.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-L2VPN-EOMPLS

**Пример**

```
esr(config-l2vpn-pw)# neighbor-address 192.168.24.78
```

**p2p**

С помощью данной команды создается L2VPN типа точка-точка (VPWS).

Использование отрицательной формы команды (no) удаляет указанный p2p.

**Синтаксис**

```
[no] p2p <NAME>
```

**Параметры**

<NAME> – имя p2p-сервиса, задается строкой до 31 символа.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-L2VPN

**Пример**

```
esr(config-l2vpn)# p2p VLAN_398
```

**pw**

С помощью данной команды создается псевдопровод.

Использование отрицательной формы команды (no) удаляет псевдопровод.

**Синтаксис**

```
pw <PW_ID> <LSR_ID>
```

```
no pw
```

## Параметры

<PW\_ID> – идентификатор pseudowire, задается в виде числа в диапазоне [1..4294967295];

<LSR\_ID> – идентификатор LSR, до которого строится pseudo-wire, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-L2VPN-EOMPLS

## Пример

```
esr(config-l2vpn-p2p)# pw 398 1.1.1.1
```

## pw-class

С помощью данной команды создается pw-class.

Использование отрицательной формы команды (no) удаляет pw-class из конфигурации.

## Синтаксис

```
[no] pw-class <WORD>
```

## Параметры

<WORD> – имя pw-class длиной [1..31] символов.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-L2VPN

## Пример

```
esr(config-l2vpn)# pw-class Class1
```

## pw-class

С помощью данной команды задается pw class для псевдопровода.

Использование отрицательной формы команды (no) убирает pw-class для данного псевдопровода.

**Синтаксис**

```
pw-class <WORD>
no pw-class
```

**Параметры**

<WORD> – имя pw-class, задается строкой длиной [1..31] символов.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-L2VPN-PW

**Пример**

```
esr(config-l2vpn-pw)# pw-class Class1
```

**transport-mode**

Данная команда позволяет выбрать транспортный режим для данного псевдопровода. Использование отрицательной формы команды (no) возвращает значение по умолчанию.

**Синтаксис**

```
transport-mode { ethernet | vlan }
no transport-mode
```

**Параметры**

ethernet – режим, при котором при входе в pseudo-wire из заголовка удаляется 802.1Q тег;  
vlan – режим, при котором 802.1Q тег может быть сохранен при передаче через pseudo-wire.

**Значение по умолчанию**

ethernet

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-L2VPN-EOMPLS

**Пример**

```
esr(config-l2vpn-p2p)# transport-mode vlan
```

**vpls**

С помощью данной команды создается L2VPN типа point-to-multipoint.

Использование отрицательной формы команды (no) удаляет указанный vpls.

**Синтаксис**

```
[no] vpls <NAME>
```

**Параметры**

<NAME> – имя vpls-сервиса, задается строкой до 31 символа.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-L2VPN

**Пример**

```
(config-l2vpn)# vpls Customer1
```

**show mpls l2vpn bindings**

Данная команда позволяет вывести информацию по параметрам псевдопровода.

**Синтаксис**

```
show mpls l2vpn binding [ { [ local <LABEL-ID> ] [ remote <LABEL ID> ] | neighbor <ADDR> | pw <PW-ID> | ve-id <VE-ID> } ]
```

**Параметры**

<LABEL-ID> – MPLS-метка, задается в виде числа в диапазоне [0..104875];

<ADDR> – IP-адрес LDP-соседа, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<PW-ID> – идентификатор pseudowire, задается в виде числа в диапазоне [1..4294967295];

<VE-ID> – идентификатор PE-маршрутизатора в домене VPLS [1.. 16384].

**Необходимый уровень привилегий**

1



## Командный режим

ROOT

### Пример

```

esr# show mpls l2vpn bindings pw 100
Neighbor: 1.1.1.1, PW ID: 100
Local label: 19
Type: Eth Tagged
Group ID: 0
MTU: 1500
Remote label: 28
Type: Eth Tagged
Group ID: 0
MTU: 1500

```

## show mpls l2vpn pseudowire

Данная команда позволяет вывести информацию по статусу псевдопровода.

### Синтаксис

```
show mpls l2vpn pseudowire [ pw <PW-ID> ] [neighbor <ADDR> ]
```

### Параметры

<PW-ID> – идентификатор pseudowire, задается в виде числа в диапазоне [1..4294967295];

<ADDR> – IP-адрес targeted LDP соседа, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

### Необходимый уровень привилегий

1

## Командный режим

ROOT

### Пример

```

esr# show mpls l2vpn pseudowire
Neighbor                               PW ID                                Type
Status
-----                               -
-----                               -
1.1.1.1                               100                                  Eth
Tagged                                Up

```

## show mpls l2vpn p2p

Данная команда позволяет получить оперативную информацию по сервису VPWS.

**Синтаксис**

```
show mpls l2vpn p2p [ name <WORD> ]
```

**Параметры**

<WORD> – имя сервиса, задается строкой длиной [1..31] символов.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример**

```
esr# sh mpls l2vpn p2p
P2P: VPWS
  gigabitethernet 1/0/4.10:
    MTU:      1500
    Status: Up
  PW ID 100, Neighbor 10.10.0.2:
    MTU:      1500
    Status TLV: Enable
    Last change (d,h:m:s): 00,00:05:35
    Status:    Up
```

**show mpls l2vpn pw-class**

Данная команда позволяет получить оперативную информацию по шаблону конфигурации, примененному к псевдопроводу.

**Синтаксис**

```
show mpls l2vpn pw-class [ name <WORD> ]
```

**Параметры**

<WORD> – имя шаблона, задается строкой длиной [1..31] символов.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример**

```
PE1# sh mpls l2vpn pw-class
```

PW-class	Neighbor	PW ID	Status	Status-tlv	MTU
PW_template	10.10.0.2	100	UP	Enable	1500

## show mpls l2vpn vpls

Данная команда позволяет получить оперативную информацию по сервису VPLS.

### Синтаксис

```
show mpls l2vpn vpls [ name <WORD> ]
```

### Параметры

<WORD> – имя сервиса, задается строкой длиной [1..31] символов.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
esr# sh mpls l2vpn vpls
VPLS: VPLS
  bridge 100:
    MTU:      1500
    Status: UP
  PWs:
    PW ID 23, Neighbor 10.10.0.1:
      MTU:      1500
      Status TLV: Enable
      Last change(d,h:m:s): 00,19:00:07
      Status:    UP
```

## 25 Резервирование

- Управление VRRP
- Управление IPv6 VRRP
- Настройка резервирования DHCP
- Настройка резервирования Firewall/NAT
- Настройка MultiWAN
- Настройка объектов отслеживания событий
- Управление Dual-Homing
- Настройка Crypto-Sync
- Настройка Cluster
- Настройка общих параметров для failover-сервисов

### Управление VRRP

- `clear vrrp-state`
- `show vrrp`
- `vrrp`
- `vrrp authentication algorithm`
- `vrrp authentication key`
- `vrrp force-up`
- `vrrp group`
- `vrrp id`
- `vrrp ip`
- `vrrp preempt delay`
- `vrrp preempt disable`
- `vrrp priority`
- `vrrp source-ip`
- `vrrp timers advertise`
- `vrrp timers garp delay`
- `vrrp timers garp refresh`
- `vrrp timers garp refresh-repeat`
- `vrrp timers garp repeat`
- `vrrp track-ip`
- `vrrp track-ip interval`
- `vrrp track-ip packets`
- `vrrp version`

#### **clear vrrp-state**

Данная команда останавливает выполнение протокола VRRP на время  $3 * \text{Advertisement\_Interval} + 1$ . Это дает возможность маршрутизатору, находящемуся в состоянии backup, выполнить перехват мастерства.

#### **Синтаксис**

```
clear vrrp-state
```

#### **Параметры**

Команда не содержит параметров.

#### **Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример 1**

```
esr# clear vrrp-state
```

**show vrrp**

Данная команда выводит информацию о протоколе VRRP.

**Синтаксис**

```
show vrrp [ <ID> ] [ vrf <VRF> ]
```

**Параметры**

<ID> – номер процесса, принимает значения [1..255];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример 1**

```
esr# show vrrp
Virtual router   Virtual IP       Priority   Preemption   State
-----
4               4.4.4.1         100      Enabled      Master
```

**Пример 2**

```
esr# show vrrp 4
Interface                bridge 50
State:                   Master
Virtual IP address:     4.4.4.1
Source IP address:      4.4.4.4
Virtual MAC address:    00:00:5e:00:01:04
Advertisement interval: 1
Preemption:             Enabled
Priority:                100
Synchronization group ID: --
```

## **vrrp**

Данная команда включает VRRP-процесс на IP-интерфейсе.

Использование отрицательной формы команды (no) выключает VRRP-процесс.

### **Синтаксис**

```
[no] vrrp
```

### **Параметры**

Команда не содержит параметров.

### **Необходимый уровень привилегий**

10

### **Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

### **Пример**

```
esr(config-if-gi)# vrrp
```

### **vrrp authentication algorithm**

Данная команда определяет алгоритм аутентификации.

Использование отрицательной формы команды (no) отключает аутентификацию.

### **Синтаксис**

```
vrrp authentication algorithm <ALGORITHM>
```

```
no vrrp authentication algorithm
```

### **Параметры**

<ALGORITHM> – алгоритм аутентификации:

- cleartext – пароль, передается открытым текстом;
- md5 – пароль, хешируется по алгоритму md5.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

**Пример**

```
esr(config-gre)# vrrp authentication algorithm cleartext
```

**vrrp authentication key**

Данная команда устанавливает пароль для аутентификации с соседом.

Использование отрицательной формы команды (no) удаляет пароль.

**Синтаксис**

```
vrrp authentication key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
```

```
no vrrp authentication key
```

**Параметры**

<CLEAR-TEXT> – ключ, задаётся строкой от 1 до 8 символов;

<ENCRYPTED-TEXT> – зашифрованный ключ размером от 1 до 8 байт (от 2 до 16 символов). Задаётся в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE

### Пример

```
esr(config-if-gi)# vrrp authentication key ascii-text 123456  
esr(config-if-gi)# vrrp authentication key ascii-text encrypted CDE650
```

### vrrp force-up

Данной командой устанавливается режим, когда виртуальный IP-интерфейс остается в состоянии UP вне зависимости от состояния самого интерфейса.

Использование отрицательной формы команды (no) устанавливает режим по умолчанию.

### Синтаксис

[no] vrrp force-up

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE

### Пример

```
esr(config-if-gi)# vrrp force-up
```



## vrrp group

Данной командой устанавливается принадлежность VRRP-интерфейса к группе. Группа предоставляет возможность синхронизировать несколько VRRP-процессов, так если в одном из процессов произойдет смена мастера, то в другом процессе также произведется смена ролей.

Использование отрицательной формы команды (no) удаляет VRRP-маршрутизатор из группы.

### Синтаксис

```
vrrp group <GRID>
```

```
no vrrp group
```

### Параметры

<GRID> – идентификатор группы VRRP-маршрутизатора, принимает значения [1..32].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

### Пример

```
esr(config-if-gi)# vrrp group 10
```

## vrrp id

Данной командой устанавливается идентификатор VRRP-маршрутизатора.

Использование отрицательной формы команды (no) удаляет идентификатор виртуального маршрутизатора.

### Синтаксис

```
vrrp id <VRID>
```

```
no vrrp id
```

**Параметры**

<VRID> – идентификатор VRRP-маршрутизатора, принимает значения [1..255].

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

**Пример**

```
esr(config-if-gi)# vrrp id 125
```

**vrrp ip**

Данной командой устанавливается виртуальный IP-адрес VRRP-маршрутизатора.

Использование отрицательной формы команды (no) удаляет виртуальный IP-адрес маршрутизатора.

**Синтаксис**

```
vrrp ip <ADDR/LEN > [ secondary ]
```

```
no vrrp ip
```

**Параметры**

<ADDR/LEN> – виртуальный IP-адрес и длина маски, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32]. Можно указать несколько IP-адресов перечислением через запятую. Может быть назначено до 8 IP-адресов на интерфейс.

secondary – ключ для установки дополнительного IP-адреса.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE

### Пример

```
esr(config-bridge)# vrrp ip 115.0.0.1
```

### vrrp preempt delay

Данной командой задается временной интервал, по истечении которого Backup-маршрутизатор с более высоким приоритетом будет пытаться перехватить на себя роль Master у текущего Master-маршрутизатора с более низким приоритетом.

Использование отрицательной формы команды (no) восстанавливает настройки по умолчанию.

### Синтаксис

```
[no] vrrp preempt delay <TIME>
```

```
no vrrp preempt delay
```

### Параметры

<TIME> – время ожидания, определяется в секундах [0..1000].

### Значение по умолчанию

0

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL

## CONFIG-BRIDGE

### Пример

```
esr(config-if-gi)# vrrp preempt delay 100
```

### vrrp preempt disable

Данной командой определяется, будет ли Backup-маршрутизатор с более высоким приоритетом пытаться перехватить на себя роль Master у текущего Master-маршрутизатора с более низким приоритетом.

Исключением является маршрутизатор, у которого виртуальный IP-адрес совпадает с IP-адресом на интерфейсе, он всегда будет перехватывать на себя роль Master вне зависимости от данной настройки.

Также исключением является равенство значений vrrp priority, маршрутизатор не будет пытаться перехватывать на себя роль Master вне зависимости от данной настройки.

Использование отрицательной формы команды (no) восстанавливает настройки по умолчанию.

### Синтаксис

```
[no] vrrp preempt disable
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Переключение включено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

### Пример

```
esr(config-if-gi)# vrrp preempt disable
```

## **vrrp priority**

Данной командой устанавливается приоритет VRRP-маршрутизатора.

Использование отрицательной формы команды (no) устанавливает значение приоритета по умолчанию.

### **Синтаксис**

```
vrrp priority <PR>
```

```
no vrrp priority
```

### **Параметры**

<PR> – приоритет VRRP-маршрутизатора, принимает значения [1..254].

### **Значение по умолчанию**

100

### **Необходимый уровень привилегий**

10

### **Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

### **Пример**

```
esr(config-if-gi)# vrrp priority 189
```

## **vrrp source-ip**

Данной командой устанавливается IP-адрес, который будет использоваться в качестве IP-адреса отправителя для VRRP-сообщений.

Использование отрицательной формы команды (no) удаляет указанный IP-адрес.

### **Синтаксис**

```
vrrp source-ip <IP>
```

```
no vrrp source-ip
```

## Параметры

<IP> – IP-адрес отправителя, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

## Пример

```
esr(config-bridge)# vrrp source-ip 115.0.0.10
```

## vrrp timers advertise

Данная команда определяет интервал между отправкой VRRP-сообщений.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

## Синтаксис

```
vrrp timers advertise <TIME>
```

```
no vrrp timers advertise
```

## Параметры

<TIME> – время в секундах, принимает значения [1..40].

## Значение по умолчанию

1

## Необходимый уровень привилегий

10

**Командный режим**

CONFIG-IF-GI  
 CONFIG-IF-TE  
 CONFIG-IF-TWE  
 CONFIG-IF-FO  
 CONFIG-IF-HU  
 CONFIG-IF-SUB  
 CONFIG-IF-QINQ  
 CONFIG-IF-PORT-CHANNEL  
 CONFIG-BRIDGE

**Пример**

```
esr(config-gre)# vrrp timers advertise 4
```

**vrrp timers garp delay**

Данная команда определяет интервал, по истечении которого происходит отправка Gratuitous ARP сообщения(ий) при переходе маршрутизатора в состояние Master.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

**Синтаксис**

```
vrrp timers garp delay <TIME>
no vrrp timers garp delay
```

**Параметры**

<TIME> – время в секундах, принимает значения [1..60].

**Значение по умолчанию**

5

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI  
 CONFIG-IF-TE  
 CONFIG-IF-TWE  
 CONFIG-IF-FO  
 CONFIG-IF-HU  
 CONFIG-IF-SUB

CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE

### Пример

```
esr(config-gre)# vrrp timers garp delay 4
```

### vrrp timers garp refresh

Данная команда определяет интервал, по истечении которого будет происходить периодическая отправка Gratuituous ARP-сообщения(ий), пока маршрутизатор находится в состоянии Master.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

### Синтаксис

```
vrrp timers garp refresh <TIME>  
no vrrp timers garp refresh
```

### Параметры

<TIME> – время в секундах, принимает значения [1..65535].

### Значение по умолчанию

Периодическая отправка отключена.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE

### Пример

```
esr(config-gre)# vrrp timers garp refresh 4
```



### **vrrp timers garp refresh-repeat**

Данная команда определяет количество Gratuitous ARP-сообщений, которые будут отправляться с периодом **garp refresh**, пока маршрутизатор находится в состоянии Master.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

#### **Синтаксис**

```
vrrp timers garp refresh-repeat <COUNT>
```

```
no vrrp timers garp refresh-repeat
```

#### **Параметры**

<COUNT> – количество сообщений, принимает значения [1..60].

#### **Значение по умолчанию**

1

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

#### **Пример**

```
esr(config-gre)# vrrp timers garp refresh-repeat 10
```

### **vrrp timers garp repeat**

Данная команда определяет количество Gratuitous ARP сообщений, которые будут отправлены при переходе маршрутизатора в состояние Master.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

#### **Синтаксис**

```
vrrp timers garp repeat <COUNT>
```

```
no vrrp timers garp repeat
```

### Параметры

<COUNT> – количество сообщений, принимает значения [1..60].

### Значение по умолчанию

5

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

### Пример

```
esr(config-gre)# vrrp timers garp repeat 10
```

### vrrp track-ip

Данная команда активирует управление статусом vrrp на основе icmp-reply от определенного IP-адреса. Использование отрицательной формы команды (no) отключает управление статусом vrrp на основе icmp-reply от определенного IP-адреса.

### Синтаксис

```
[no] vrrp track-ip <ADDR>
```

### Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

### Значение по умолчанию

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

**Пример**

```
esr(config-if)# vrrp track-ip 192.168.154.22
```

**vrrp track-ip interval**

Данная команда устанавливает интервал проверки при помощи icmp-request для управления статусом vrrp на основе icmp-reply от определенного IP-адреса.

Использование отрицательной формы команды (no) устанавливает значение интервала по умолчанию.

**Синтаксис**

```
vrrp track-ip interval <TIME>
```

```
no vrrp track-ip interval
```

**Параметры**

<TIME> – интервал отправки icmp-request в секундах [3..60].

**Значение по умолчанию**

10

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE

### Пример

```
esr(config-if)# vrrp track-ip interval 20
```

### vrrp track-ip packets

Данная команда устанавливает количество icmp-request при каждой проверке для управления статусом vrrp на основе icmp-reply от определенного IP-адреса.

Использование отрицательной формы команды (no) устанавливает значение количества пакетов по умолчанию.

### Синтаксис

```
vrrp track-ip packets <COUNT>  
no vrrp track-ip packets
```

### Параметры

<COUNT> – интервал отправки icmp-request в секундах [1..5].

### Значение по умолчанию

5

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE

## Пример

```
esr(config-if)# vrrp track-ip packets 3
```

## vrrp version

Данной командой задаётся версия VRRP-протокола.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
vrrp version <VERSION>
```

```
no vrrp version
```

## Параметры

<VERSION> – версия VRRP-протокола: 2, 3.

## Значение по умолчанию

2

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

## Пример

```
esr(config-gre)# vrrp version 3
```

## Управление IPv6 VRRP

- [ipv6 vrrp](#)
- [ipv6 vrrp group](#)
- [ipv6 vrrp id](#)
- [ipv6 vrrp ip](#)
- [ipv6 vrrp preempt delay](#)
- [ipv6 vrrp preempt disable](#)
- [ipv6 vrrp priority](#)
- [ipv6 vrrp source-ip](#)
- [ipv6 vrrp timers advertise](#)
- [ipv6 vrrp timers nd delay](#)
- [ipv6 vrrp timers nd refresh](#)
- [ipv6 vrrp timers nd refresh-repeat](#)
- [ipv6 vrrp timers nd repeat](#)
- [show ipv6 vrrp](#)

### ipv6 vrrp

Данная команда включает VRRP-процесс на IPv6-интерфейсе.

Использование отрицательной формы команды (no) выключает VRRP-процесс.

### Синтаксис

```
[no] ipv6 vrrp
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

### Пример

```
esr(config-if-gi)# ipv6 vrrp
```

## ipv6 vrrp group

Данной командой устанавливается принадлежность VRRP-маршрутизатора к группе. Группа предоставляет возможность синхронизировать несколько VRRP-процессов. Так, если в одном из процессов произойдет смена мастера, то в другом процессе также произведется смена ролей.

Использование отрицательной формы команды (no) удаляет VRRP-маршрутизатор из группы.

### Синтаксис

```
ipv6 vrrp group <GRID>
```

```
no ipv6 vrrp group
```

### Параметры

<GRID> – идентификатор группы VRRP-маршрутизатора, принимает значения [1..32].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

### Пример

```
esr(config-if-gi)# ipv6 vrrp group 10
```

## ipv6 vrrp id

Данной командой устанавливается идентификатор VRRP-маршрутизатора.

Использование отрицательной формы команды (no) удаляет идентификатор виртуального маршрутизатора.

### Синтаксис

```
ipv6 vrrp id <VRID>
```

```
no ipv6 vrrp id
```

**Параметры**

<VRID> – идентификатор VRRP-маршрутизатора, принимает значения [1..255].

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

**Пример**

```
esr(config-if-gi)# ipv6 vrrp id 125
```

**ipv6 vrrp ip**

Данной командой устанавливается виртуальный IPv6-адрес VRRP-маршрутизатора.

Использование отрицательной формы команды (no) удаляет виртуальный IPv6-адрес маршрутизатора.

**Синтаксис**

```
ipv6 vrrp ip <IPV6-ADDR> [secondary]
```

```
no ipv6 vrrp ip
```

**Параметры**

<IPV6-ADDR> – виртуальный IPv6-адрес, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF]. Можно указать несколько IPv6-адресов перечислением через запятую. Может быть назначено до 8 IPv6-адресов на интерфейс;

secondary – ключ для установки дополнительного IP-адреса.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI



CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE

### Пример

```
esr(config-bridge)# ipv6 vrrp ip fc00::1
```

### ipv6 vrrp preempt delay

Данной командой задается временной интервал, по истечении которого Backup-маршрутизатор с более высоким приоритетом будет пытаться перехватить на себя роль Master у текущего Master-маршрутизатора с более низким приоритетом.

Использование отрицательной формы команды (no) восстанавливает настройки по умолчанию.

### Синтаксис

```
ipv6 vrrp preempt delay <TIME>
```

```
no ipv6 vrrp preempt delay
```

### Параметры

<TIME> – время ожидания, определяется в секундах [0..1000].

### Значение по умолчанию

0

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL

## CONFIG-BRIDGE

**Пример**

```
esr(config-if-gi)# ipv6 vrrp preempt delay 100
```

**ipv6 vrrp preempt disable**

Данной командой определяется, будет ли Backup-маршрутизатор с более высоким приоритетом пытаться перехватить на себя роль Master у текущего Master-маршрутизатора с более низким приоритетом.

Исключением является маршрутизатор, у которого виртуальный IPv6-адрес совпадает с IPv6-адресом на интерфейсе – он всегда будет перехватывать на себя роль Master вне зависимости от данной настройки.

Использование отрицательной формы команды (no) восстанавливает настройки по умолчанию.

**Синтаксис**

```
[no] ipv6 vrrp preempt disable
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Переключение включено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

**Пример**

```
esr(config-if-gi)# ipv6 vrrp preempt disable
```

## ipv6 vrrp priority

Данной командой устанавливается приоритет VRRP-маршрутизатора.

Использование отрицательной формы команды (no) устанавливает значение приоритета по умолчанию.

### Синтаксис

```
ipv6 vrrp priority <PR>  
no ipv6 vrrp priority
```

### Параметры

<PR> – приоритет VRRP-маршрутизатора, принимает значения [1..254].

### Значение по умолчанию

100

### Необходимый уровень привилегий

10

### Командный режим

```
CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE
```

### Пример

```
esr(config-if-gi)# ipv6 vrrp priority 189
```

## ipv6 vrrp source-ip

Данной командой устанавливается IPv6-адрес, который будет использоваться в качестве IPv6-адреса отправителя для VRRP-сообщений.

Использование отрицательной формы команды (no) удаляет указанный IPv6-адрес.

### Синтаксис

```
ipv6 vrrp source-ip <IPV6-ADDR>  
no ipv6 vrrp source-ip
```

## Параметры

<IPv6-ADDR> – IPv6-адрес отправителя, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

## Пример

```
esr(config-bridge)# ipv6 vrrp source-ip fc00::2
```

## ipv6 vrrp timers advertise

Данная команда определяет интервал между отправкой VRRP-сообщений.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

## Синтаксис

```
ipv6 vrrp timers advertise <TIME>
```

```
no ipv6 vrrp timers advertise
```

## Параметры

<TIME> – время в секундах, принимает значения [1..40].

## Значение по умолчанию

1

## Необходимый уровень привилегий

10

**Командный режим**

CONFIG-IF-GI  
 CONFIG-IF-TE  
 CONFIG-IF-TWE  
 CONFIG-IF-FO  
 CONFIG-IF-HU  
 CONFIG-IF-SUB  
 CONFIG-IF-QINQ  
 CONFIG-IF-PORT-CHANNEL  
 CONFIG-BRIDGE

**Пример**

```
esr(config-if-gi)# ipv6 vrrp timers advertise 4
```

**ipv6 vrrp timers nd delay**

Данная команда определяет задержку между установлением IPv6 VRRP состояния MASTER и началом рассылки ND-сообщений.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

**Синтаксис**

```

ipv6 vrrp timers nd delay <TIME>
no ipv6 vrrp timers nd delay

```

**Параметры**

<TIME> – время в секундах, принимает значения [1..60].

**Значение по умолчанию**

5

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI  
 CONFIG-IF-TE  
 CONFIG-IF-TWE  
 CONFIG-IF-FO  
 CONFIG-IF-HU  
 CONFIG-IF-SUB

CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE

### Пример

```
esr(config-if-gi)# ipv6 vrrp timers nd delay 30
```

### ipv6 vrrp timers nd refresh

Данная команда определяет период обновления информации протокола ND для IPv6 VRRP в состоянии MASTER.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

### Синтаксис

```
ipv6 vrrp timers nd refresh <TIME>
```

```
no ipv6 vrrp timers nd refresh
```

### Параметры

<TIME> – время в секундах, принимает значения [1..65535].

### Значение по умолчанию

5

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE

### Пример

```
esr(config-if-gi)# ipv6 vrrp timers nd refresh 32768
```

### ipv6 vrrp timers nd refresh-repeat

Данная команда определяет количество ND-сообщений, отправляемых за период обновления для IPv6 VRRP в состоянии MASTER.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

#### Синтаксис

```
ipv6 vrrp timers nd refresh-repeat <NUM>  
no ipv6 vrrp timers nd refresh-repeat
```

#### Параметры

<NUM> – количество, принимает значения [1..60].

#### Значение по умолчанию

0

#### Необходимый уровень привилегий

10

#### Командный режим

```
CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE
```

#### Пример

```
esr(config-if-gi)# ipv6 vrrp timers nd refresh-repeat 12
```

### ipv6 vrrp timers nd repeat

Данная команда определяет количество отправок ND-пакетов после установки IPv6 VRRP в состояние MASTER.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

### Синтаксис

```
ipv6 vrrp timers nd repeat <NUM>  
no ipv6 vrrp timers nd repeat
```

### Параметры

<NUM> – количество, принимает значения [1..60].

### Значение по умолчанию

1

### Необходимый уровень привилегий

10

### Командный режим

```
CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE
```

### Пример

```
esr(config-if-gi)# ipv6 vrrp timers nd repeat 4
```

### show ipv6 vrrp

Данная команда выводит информацию о протоколе VRRP.

### Синтаксис

```
show ipv6 vrrp [ <ID> ] [ vrf <VRF> ]
```

### Параметры

<ID> – номер процесса, принимает значения [1..255];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.



**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример 1**

```

esr# show ipv6 vrrp
Virtual router   Virtual IP           Priority   Preemption
State
-----
-----
2               fc00::3             100       Disabled
Master

```

**Пример 2**

```

esr# show ipv6 vrrp 2
Interface          gi1/0/1
State:             Master
Source IP address: fe80::aaf9:4bff:feaa:3a1
Virtual IP address: fc00::3
Virtual MAC address: 00:00:5e:00:01:02
Advertisement interval: 1
Preemption:        Disabled
Priority:           100
Synchronization group ID: --

```

## Настройка резервирования DHCP

- [auto-partner-down](#)
- [enable](#)
- [ip dhcp-server failover](#)
- [maximum-client-lead-time](#)
- [mode](#)
- [role](#)
- [show ip dhcp server failover](#)

### auto-partner-down

Данной командой устанавливается время отработки DHCP-lease, при котором аренда может быть продлена одним из узлов отказоустойчивости, не связываясь с другим.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

### Синтаксис

```
ip dhcp-server failover auto-partner-down <TIME>
no ip dhcp-server failover auto-partner-down
```

### Параметры

<TIME> – время, определяется в секундах [1..86400].

### Значение по умолчанию

0

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-DHCP-SERVER-FAILOVER

### Пример

```
esr(config-dhcp-server-failover)# auto-partner-down 200
```

### enable

Данной командой включается работа резервирования DHCP-сервера.

Использование отрицательной формы команды (no) выключает работа резервирования DHCP-сервера.

### Синтаксис

```
[no] enable
```

**Параметры**

Отсутствуют.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config-dhcp-server-failover)# enable
```

**ip dhcp-server failover**

Данной командой осуществляется переход в режим конфигурирования резервирования DHCP-сервера. Использование отрицательной формы команды (no) удаляет настройки резервирования DHCP-сервера.

**Синтаксис**

```
[no] ip dhcp-server failover [ vrf <VRF> ] [ all ]
```

**Параметры**

<VRF> – имя VRF, задается строкой до 31 символа;

all – при использовании ключа при удалении, удаляется конфигурация резервирования DHCP-серверов во всех VRF и глобальной конфигурации.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip dhcp-server failover
esr(config-dhcp-server-failover)#
```

## maximum-client-lead-time

Данной командой устанавливается максимальное время, на которое DHCP-сервер может продлить время аренды IP-адреса.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
maximum-client-lead-time <TIME>
```

```
no maximum-client-lead-time
```

### Параметры

<TIME> – время, определяется в секундах [1..86400].

### Значение по умолчанию

1800

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-DHCP-SERVER-FAILOVER

### Пример

```
esr(config-dhcp-server-failover)# maximum-client-lead-time 200
```

## mode

Данной командой настраивается режим работы DHCP failover.

Использование отрицательной формы команды (no) устанавливает режим работы DHCP failover по умолчанию.

### Синтаксис

```
mode { active-active | active-standby }
```

```
no mode
```

### Значение по умолчанию

Active-Active

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-DHCP-SERVER-FAILOVER

### Пример

```
esr(config-dhcp-server-failover)# mode active-standby
```

### role

Данной командой определяется роль DHCP-сервера при работе в режиме резервирования Active-Active.

Использование отрицательной формы команды (no) удаляет установленную роль DHCP-сервера при работе в режиме резервирования Active-Active.

### Синтаксис

```
role <ROLE>
```

### Параметры

<ROLE> – роль DHCP-сервера при работе в режиме резервирования:

- primary – режим активного DHCP-сервера;
- secondary – режим резервного DHCP-сервера.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-DHCP-SERVER-FAILOVER

### Пример

```
esr(config-dhcp-server-failover)# role primary
```

### show ip dhcp server failover

Данная команда позволяет посмотреть состояние резервирования DHCP-сервера.

### Синтаксис

```
show ip dhcp server failover [ vrf <VRF> | all ]
```

### Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример**

```
esr# show ip dhcp server failover
VRF:      --
State: Successful
```

## Настройка резервирования Firewall/NAT

- [clear ip firewall failover counters](#)
- [enable](#)
- [ip firewall failover](#)
- [port](#)
- [show ip firewall failover](#)
- [show ip firewall sessions failover](#)
- [show ip nat translations failover](#)
- [sync-type](#)

### clear ip firewall failover counters

Данная команда позволяет очистить счетчики работы firewall failover.

#### Синтаксис

```
clear ip firewall failover counters
```

#### Необходимый уровень привилегий

10

#### Командный режим

ROOT

#### Пример

```
esr# clear ip firewall failover counters
```

### enable

Данной командой включается работа резервирования firewall.

Использование отрицательной формы команды (no) отключает резервирование firewall.

#### Синтаксис

```
[no] enable
```

#### Параметры

Отсутствуют.

#### Значение по умолчанию

Отключено.

#### Необходимый уровень привилегий

10

## Командный режим

CONFIG-FIREWALL-FAILOVER

### Пример

```
esr(config-firewall-failover)# enable
```

### ip firewall failover

Данной командой осуществляется переход в режим конфигурирования резервирования сессий Firewall. Использование отрицательной формы команды (no) удаляет настройки резервирования сессий Firewall.

### Синтаксис

```
[no] ip firewall failover
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG

### Пример

```
esr(config)# ip firewall failover
```

### port

Данной командой устанавливается номер UDP-порта службы резервирования сессий Firewall, через который происходит обмен информацией при работе в unicast-режиме.

Использование отрицательной формы команды (no) удаляет номер порта службы резервирования сессий Firewall.

### Синтаксис

```
port <PORT>
```

```
no port
```



## Параметры

<PORT> – номер порта службы резервирования сессий Firewall, указывается в диапазоне [1..65535].

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-FIREWALL-FAILOVER

## Пример

```
esr(config-firewall-failover)# port 3333
```

## show ip firewall failover

Данная команда позволяет посмотреть состояние резервирования сессий Firewall.

## Синтаксис

```
show ip firewall failover
```

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```

esr# show ip firewall failover
Communication interface:                port-channel 1
Status:                                           Running
Bytes sent:                                       17184
Bytes received:                                  13088
Packets sent:                                    866
Packets received:                               817
Send errors:                                     0
Receive errors:                                 0
Internal sessions cache counters:
  Active entries:                                4
  Added:                                         47
  Deleted:                                      43
  Updated:                                      13
  Failed adding:                                0

```

```

    No memory left:                0
    No space left:                 0
Failed deleting:                  0
    No entry found:                0
Failed updating:                  0
    No entry found:                0
External sessions cache counters:
  Active entries:                 0
  Added:                          1
  Deleted:                        6
  Updated:                        0
  Installed to Kernel:            4
  Failed adding:                  0
    No memory left:                0
    No space left:                 0
  Failed deleting:                0
    No entry found:                0
  Failed updating:                0
    No entry found:                0
  Failed installing to Kernel:     0

```

### show ip firewall sessions failover

Данная команда используется для просмотра кэшей IP-сессий при работе firewall failover.

#### Синтаксис

```

show ip firewall sessions failover { internal | external } [ protocol <TYPE> ] [ inside-
source-address <ADDR> ] [ outside-source-address <ADDR> ] [ inside-destination-address
<ADDR> ] [ outside-destination-address <ADDR> ] [ inside-source-port <PORT> ] [ outside-
source-port <PORT> ] [ inside-destination-port <PORT> ] [ outside-destination-port
<PORT> ] [ summary ] [ configuration ] [ expected ]

```

#### Параметры

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<PORT> – TCP/UDP-порт, принимает значения [1..65535];

internal – кэш синхронизации с ядром.

external – кэш синхронизации с соседом.

inside-source-address – команда для указания IP-адреса источника входящих пакетов;

inside-destination-address – команда для указания IP-адреса назначения входящих пакетов;

outside-source-address – команда для указания IP-адреса источника отправляемых пакетов;

outside-destination-address – команда для указания IP-адреса назначения отправляемых пакетов;

inside-source-port – ключ для указания TCP/UDP-порта отправителя в входящих пакетах;

outside-source-port – ключ для указания TCP/UDP-порта отправителя в отправляемых пакетах;

inside-destination-port – ключ для указания TCP/UDP-порта назначения в входящих пакетах;

outside-destination-port – ключ для указания TCP/UDP-порта назначения в отправляемых пакетах;

summary – выводит суммарную статистику по IP-сессиям.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример**

```

esr# show ip firewall sessions failover internal
Codes: E - expected, U - unreplied,
      A - assured, C - confirmed

Prot   Inside source      Inside destination  Outside source      Outside
destination      Pkts
-----
-----
udp    10.10.10.2:500     10.10.10.1:500     10.10.10.2:500     10.10.10.1:500
--

```

**show ip nat translations failover**

Данная команда используется для просмотра кэша IP-сессий трансляции при работе nat failover.

**Синтаксис**

```

show ip nat translations failover { internal | external } [ protocol <TYPE> ] [ inside-
source-address <ADDR> ] [ outside-source-address <ADDR> ] [ inside-destination-address
<ADDR> ] [ outside-destination-address <ADDR> ] [ inside-source-port <PORT> ] [ outside-
source-port <PORT> ] [ inside-destination-port <PORT> ] [ outside-destination-port
<PORT> ] [ summary ]

```

## Параметры

summary – выводит суммарную статистику по сессиям трансляции;

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<PORT> – TCP/UDP-порт, принимает значения [1..65535].

Для Source NAT:

- inside-source-address – ключ для указания IP-адреса источника до трансляции;
- inside-destination-address – ключ для указания IP-адреса назначения на входе в маршрутизатор;
- outside-source-address – ключ для указания IP-адреса источника после трансляции;
- outside-destination-address – ключ для указания IP-адреса назначения на выходе из маршрутизатора.
- inside-source-port – ключ для указания TCP/UDP-порта отправителя до трансляции;
- outside-source-port – ключ для указания TCP/UDP-порта отправителя после трансляции;
- inside-destination-port – ключ для указания TCP/UDP-порта назначения до трансляции;
- outside-destination-port – ключ для указания TCP/UDP-порта назначения после трансляции.

Для Destination NAT:

- inside-source-address – ключ для указания IP-адреса источника на выходе из маршрутизатора;
- inside-destination-address – ключ для указания IP-адреса назначения после трансляции;
- outside-source-address – ключ для указания IP-адреса источника на входе в маршрутизатор;
- outside-destination-address – ключ для указания IP-адреса назначения до трансляции;
- inside-source-port – ключ для указания TCP/UDP-порта отправителя до трансляции;
- outside-source-port – ключ для указания TCP/UDP-порта отправителя после трансляции;
- inside-destination-port – ключ для указания TCP/UDP-порта назначения до трансляции;
- outside-destination-port – ключ для указания TCP/UDP-порта назначения после трансляции.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```

esr# show ip nat translations failover internal
Prot  Inside source          Inside destination      Outside source          Outside
destination            Pkts
-----
-----
tcp    192.168.10.10:39268     217.134.15.73:22       222.222.222.2:39268    217.134.15.73:22
--

```

## sync-type

Данной командой определяется режим обмена информацией между основным и резервным маршрутизаторами.

Использование отрицательной формы команды (no) удаляет режим работы резервирования Firewall.

**Синтаксис**

```
sync-type <MODE>
```

```
no sync-type
```

**Параметры**

<MODE> – режим обмена информацией:

- unicast – режим unicast;
- multicast – режим multicast.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-FIREWALL-FAILOVER

**Пример**

```
esr(config-firewall-failover)# sync-type multicast
```

## Настройка MultiWAN

- `description`
- `enable`
- `extended count`
- `extended jitter`
- `extended loss`
- `extended period`
- `extended rtt`
- `extended timeout`
- `failover`
- `ip address`
- `ipv6 address`
- `ipv6 wan load-balance enable`
- `ipv6 wan load-balance failure-count`
- `ipv6 wan load-balance nexthop`
- `ipv6 wan load-balance rule`
- `ipv6 wan load-balance source-address`
- `ipv6 wan load-balance success-count`
- `ipv6 wan load-balance target-list`
- `ipv6 wan load-balance target-list`
- `ipv6 wan load-balance target-list check-all`
- `mode`
- `outbound`
- `resp-time`
- `target`
- `stickiness`
- `wan load-balance enable`
- `wan load-balance failure-count`
- `wan load-balance nexthop`
- `wan load-balance rule`
- `wan load-balance source-address`
- `wan load-balance success-count`
- `wan load-balance target-list`
- `wan load-balance target-list`
- `wan load-balance target-list check-all`
- `show wan rules`
- `show wan cellular status modem`
- `show wan interfaces status`
- `show wan tunnels status`
- `show ipv6 wan rules`
- `show ipv6 wan interfaces status`

### description

Данной командой определяется описание правила.

Использование отрицательной формы команды (`no`) удаляет описание.

### Синтаксис

```
description <DESCRIPTION>
```

```
no description
```

## Параметры

<DESCRIPTION> – описание правила WAN, задаётся строкой до 255 символов.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WAN-RULE

CONFIG-IPV6-WAN-RULE

CONFIG-WAN-TARGET

CONFIG-IPV6-WAN-TARGET

## Пример

```
esr(config-wan-rule)# description "tunnel to branch"
```

## enable

Данной командой включаются правило WAN, проверка цели.

Использование отрицательной формы команды (no) выключает правило WAN, проверку цели.

## Синтаксис

[no] enable

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Выключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WAN-RULE

CONFIG-WAN-TARGET

CONFIG-IPV6-WAN-RULE

CONFIG-IPV6-WAN-TARGET

## Пример

Проверка цели:

```
esr(config-ipv6-wan-rule)# enable
```

## extended count

Данной командой определяется количество одновременно отправленных ICMP-запросов. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
extended count <NUM>
```

```
no extended count
```

## Параметры

<NUM> – количество одновременно отправленных ICMP-запросов, принимает значение в диапазоне [10..100].

## Значение по умолчанию

10

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WAN-TARGET

## Пример

```
esr(config-wan-target)# extended count 75
```

## extended jitter

Данная команда определяет пороговое значение джиттера, при превышении которого качество канала является неудовлетворительным.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
extended jitter <TIME>
```

```
no extended jitter
```



## Параметры

<TIME> – пороговое значение джиттера в миллисекундах, принимает значение в диапазоне [50-1000].

## Значение по умолчанию

50

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WAN-TARGET

## Пример

```
esr(config-wan-target)# extended jitter 78500
```

## extended loss

Данная команда определяет пороговое значение процента потерь, при превышении которого качество канала является неудовлетворительным.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
extended loss <NUM>
```

```
no extended loss
```

## Параметры

<NUM> – пороговое значение процента потерь [1..100].

## Значение по умолчанию

1

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WAN-TARGET

## Пример

```
esr(config-wan-target)# extended loss 80
```

### extended period

Данная команда определяет период отправки ICMP-запросов.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### Синтаксис

```
extended period <TIME>
```

```
no extended period
```

#### Параметры

<TIME> – период отправки ICMP запросов в секундах, принимает значение в диапазоне [10..600].

#### Значение по умолчанию

240

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-WAN-TARGET

#### Пример

```
esr(config-wan-target)# extended period 120
```

### extended rtt

Данная команда определяет максимальное пороговое значение круговой задержки, при превышении которого качество канала является неудовлетворительным.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### Синтаксис

```
extended rtt <TIME>
```

```
no extended rtt
```

#### Параметры

<TIME> – максимальное пороговое значение RTT в миллисекундах, принимает значение в диапазоне [200..1000].

#### Значение по умолчанию

240

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WAN-TARGET

**Пример**

```
esr(config-wan-target)# extended rtt 800
```

**extended timeout**

Данная команда определяет максимальное время ожидания ответа на ICMP-запрос. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
extended timeout <TIME>
```

```
no extended timeout
```

**Параметры**

<TIME> – максимальное время ожидания ответа на ICMP-запрос в миллисекундах, принимает значение в диапазоне [500..10000].

**Значение по умолчанию**

1000

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WAN-TARGET

**Пример**

```
esr(config-wan-target)# extended timeout 5000
```

**failover**

Данной командой осуществляется переключение из режима балансировки в режим резервирования. Использование отрицательной формы команды (no) возвращает режим балансировки.

**Синтаксис**

```
[no] failover
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WAN-RULE

CONFIG-IPV6-WAN-RULE

**Пример**

```
esr(config-wan-rule)# failover
```

**ip address**

Данной командой указывается IP-адрес проверки.

Использование отрицательной формы команды (no) удаляет указанный адрес.

**Синтаксис**

```
ip address <ADDR>
```

```
no ip address
```

**Параметры**

<ADDR> – IP-адрес назначения, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-WAN-TARGET

**Пример**

```
esr(config-wan-target)# ip address 18.168.1.25
```

## ipv6 address

Данной командой указывается IPv6-адрес проверки.

Использование отрицательной формы команды (no) удаляет указанный адрес.

### Синтаксис

```
ipv6 address <IPV6-ADDR>
```

```
no ipv6 address
```

### Параметры

<IPV6-ADDR> – IPv6-адрес назначения, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IPV6-WAN-TARGET

### Пример

```
esr(config-ipv6-wan-target)# ipv6 address fc00::2
```

## ipv6 wan load-balance enable

Данной командой включается WAN режим на интерфейсе для IPv6-стека.

Использование отрицательной формы команды (no) выключает WAN-режим для IPv6-стека.

### Синтаксис

```
[no] ipv6 wan load-balance enable
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено

### Необходимый уровень привилегий

10

**Командный режим**

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-SERIAL  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-LT

**Пример**

```
esr(config-if-gi)# ipv6 wan load-balance enable
```

**ipv6 wan load-balance failure-count**

Данной командой определяется количество неудачных попыток проверки соединения через IPv6-стек, после которых, при отсутствии ответа от встречной стороны, соединение считается неактивным.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
ipv6 wan load-balance failure-count <VALUE>  
no ipv6 wan load-balance failure-count
```

**Параметры**

<VALUE> – количество попыток, определяется в диапазоне [1...10].

**Значение по умолчанию**

5

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE

CONFIG-IF-FO  
 CONFIG-IF-HU  
 CONFIG-IF-SUB  
 CONFIG-IF-QINQ  
 CONFIG-IF-PORT-CHANNEL  
 CONFIG-BRIDGE  
 CONFIG-LT

### Пример

```
esr(config-if-gi)# ipv6 wan load-balance failure-count 3
```

### ipv6 wan load-balance nexthop

Данной командой определяется IPv6-адрес соседа, который будет указан в качестве одного из шлюзов в статическом маршруте, создаваемом службой MultiWAN.

Использование отрицательной формы команды (no) удаляет указанный IPv6-адрес соседа.

### Синтаксис

```
ipv6 wan load-balance nexthop <IPV6-ADDR>
```

```
no ipv6 wan load-balance nexthop
```

### Параметры

<IPV6-ADDR> – IPv6-адрес назначения (шлюз), задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
 CONFIG-IF-TE  
 CONFIG-IF-TWE  
 CONFIG-IF-FO  
 CONFIG-IF-HU  
 CONFIG-IF-SUB  
 CONFIG-IF-QINQ  
 CONFIG-IF-PORT-CHANNEL  
 CONFIG-BRIDGE  
 CONFIG-LT

## Пример

```
esr(config-if-gi)#ipv6 wan load-balance nexthop 220::71
```

### ipv6 wan load-balance rule

Данной командой создается правило WAN и осуществляется переход в режим настройки параметров правила для протокола IPv6.

Использование отрицательной формы команды (no) удаляет созданное WAN-правило.

### Синтаксис

```
ipv6 wan load-balance rule <ID>
no ipv6 wan load-balance rule { <ID> | all }
```

### Параметры

<ID> – идентификатор создаваемого правила, принимает значения:

- Для ESR-10/12V/12VF/15/15R/15VF – [1..25];
- Для ESR-20/21/30/31/100/200/1000/1200/1500/1511/1700/3100/3200/3200L/3300 – [1..50].

Значение «all» используется при удалении всех правил WAN для протокола IPv6.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

## Пример

```
esr(config)# ipv6 wan load-balance rule 1
```

### ipv6 wan load-balance source-address

Данной командой определяется IPv6-адрес маршрутизатора, который будет использоваться в качестве IPv6-адреса источника в отправляемых ICMPv6-пакетах для проверки целей.

Использование отрицательной формы команды (no) удаляет указанный IPv6-адрес источника.

### Синтаксис

```
ipv6 wan load-balance source-address <IPV6-ADDR>
no ipv6 wan load-balance source-address
```

### Параметры

<IPV6-ADDR> – IPv6-адрес источника, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].



**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-LT

**Пример**

```
esr(config-if-gi)#ipv6 wan load-balance source-address 220::71
```

**ipv6 wan load-balance success-count**

Данной командой определяется количество успешных попыток проверки соединения по протоколу IPv6, после которых, в случае успеха, соединение считается вновь активным.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
ipv6 wan load-balance success-count <VALUE>
```

```
no ipv6 wan load-balance success-count
```

**Параметры**

<VALUE> – количество попыток, определяется в диапазоне [1...10].

**Значение по умолчанию**

1

**Необходимый уровень привилегий**

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LT

## Пример

```
esr(config-if-gi)# ipv6 wan load-balance success-count 3
```

## ipv6 wan load-balance target-list

Данной командой создается список IPv6-адресов для проверки целостности соединения и осуществляется переход в режим настройки параметров списка.

Использование отрицательной формы команды (no) удаляет созданный список.

## Синтаксис

```
ipv6 wan load-balance target-list <NAME>
```

```
no ipv6 wan load-balance target-list { <NAME> | all }
```

## Параметры

<NAME> – название списка, задается строкой до 31 символа. Значение «all» используется при удалении всех списков IPv6-адресов для проверки целостности соединения.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# ipv6 wan load-balance target-list ten1
```

## ipv6 wan load-balance target-list

Данной командой производится привязка списка IPv6-адресов для проверки целостности соединения на сетевом интерфейсе.

Использование отрицательной формы команды (no) удаляет список с интерфейса.

### Синтаксис

```
ipv6 wan load-balance target-list <NAME>
```

no ipv6 wan load-balance target-list (удаление привязки списка в режиме конфигурирования сетевых интерфейсов)

### Параметры

<NAME> – название списка, задается строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LT

### Пример

```
esr(config)# ipv6 wan load-balance target-list ten1
```

## ipv6 wan load-balance target-list check-all

Данной командой будут проверяться все IPv6-адреса из списка проверки целостности.

Использование отрицательной формы команды (no) отменяет проверку всех IPv6-адресов из списка проверки целостности. В случае недоступности одного из проверяемых узлов, шлюз будет считаться недоступным.

### Синтаксис

```
[no] ipv6 wan load-balance target-list check-all
```

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LT

## Пример

```
esr(config-if-te)# ipv6 wan load-balance target-list check-all
```

## mode

Данная команда определяет режим проверки доступности цели для проверки состояния канала. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
mode { basic | extended }
```

```
no mode
```

## Параметры

basic – базовый, регулируется допустимым значением положительных ответов на ICMP-запросы;

extended – расширенный метод, в нем производится оценка уровня потерь, задержки, джиттер.

## Значение по умолчанию

basic

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WAN-TARGET

## Пример

```
esr(config-wan-target)# mode extended
```

## outbound

Данной командой определяются интерфейсы или туннели, которые являются шлюзами в маршруте, создаваемом службой MultiWAN. Количество шлюзов в маршруте зависит от режима работы MultiWAN:

- при балансировке в список nexthop-маршрута добавляются IP-адреса шлюзов (раздел [wan load-balance nexthop](#)) всех активных интерфейсов;
- при резервировании в качестве nexthop-маршрута выбирается IP-адрес шлюза (раздел [wan load-balance nexthop](#)) активного интерфейса с наибольшим весом.

Использование отрицательной формы команды (no) исключает указанный интерфейс или туннель из правила MultiWAN.

## Синтаксис

```
[no] outbound { interface <IF> | tunnel <TUN> } [<WEIGHT> ]
```

## Параметры

<IF> – интерфейс, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<TUN> – имя туннеля, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#) (только для IPv4);

<WEIGHT> – вес туннеля или интерфейса, определяется в диапазоне [1..255]. Если установить значение 2, то по данному интерфейсу будет передаваться в 2 раза больше трафика, чем по интерфейсу с дефолтным значением. В режиме резервирования активным будет маршрут с наибольшим весом.

## Значение по умолчанию

WEIGHT – 1

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WAN-RULE

CONFIG-IPV6-WAN-RULE

## Пример 1

```
esr(config-wan-rule)# outbound interface gigabitethernet 1/0/15
```

## Пример 2

```
esr(config-ipv6-wan-rule)# outbound interface bridge 2
```

### resp-time

Данной командой определяется время ожидания ответа на запрос по протоколу ICMP.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
resp-time <TIME>
```

```
no resp-time
```

### Параметры

<TIME> – время ожидания, определяется в секундах [1..30].

### Значение по умолчанию

5

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WAN-TARGET

CONFIG-IPV6-WAN-TARGET

## Пример

```
esr(config-wan-target)# resp-time 3
```

### target

Данной командой создается цель проверки и осуществляется переход в режим настройки параметров цели.

Использование отрицательной формы команды (no) удаляет созданную цель.

### Синтаксис

```
target <ID>
```

```
no target { <ID> | all }
```

## Параметры

<ID> – идентификатор цели, задаётся в пределах [1..50]. Если при удалении используется значение параметра «all», то будут удалены все цели для конфигулируемого списка целей.

all – удалить все цели.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WAN-TARGET-LIST

CONFIG-IPV6-WAN-TARGET-LIST

## Пример

```
esr(config-target-list)# target 1
```

## stickiness

Данной командой включается отправка ответных пакетов сессии через тот же интерфейс, через который получены входящие пакеты сессии.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

stickiness

no stickiness

## Значение по умолчанию

Выключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WAN-RULE

## Пример

```
esr(config-wan-rule)# stickiness
```

## wan load-balance enable

Данной командой включается WAN-режим на интерфейсе для IPv4-стека.

Использование отрицательной формы команды (no) выключает WAN-режим для IPv4-стека.

### Синтаксис

```
[no] wan load-balance enable
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Выключено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-F0

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-VTI

CONFIG-LT

CONFIG-CELLULAR-MODEM

CONFIG-L2TP

CONFIG-OPENVPN

CONFIG-PPPOE

CONFIG-PPTP

### Пример

```
esr(config-if-gi)# wan load-balance enable
```



## wan load-balance failure-count

Данной командой определяется количество неудачных попыток проверки соединения, после которых, при отсутствии ответа от встречной стороны, соединение считается неактивным.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
wan load-balance failure-count <VALUE>
```

```
no wan load-balance failure-count
```

### Параметры

<VALUE> – количество попыток, определяется в диапазоне [1...10].

### Значение по умолчанию

5

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-VTI

CONFIG-LT

CONFIG-CELLULAR-MODEM

CONFIG-L2TP

CONFIG-OPENVPN

CONFIG-PPPOE

## CONFIG-PPTP

**Пример**

```
esr(config-if-gi)# wan load-balance failure-count 3
```

**wan load-balance nexthop**

Данной командой определяется IP-адрес соседа, который будет указан в качестве одного из шлюзов в статическом маршруте, создаваемом службой MultiWAN.

Использование отрицательной формы команды (no) удаляет указанный IP-адрес соседа.

**Синтаксис**

```
wan load-balance nexthop { <ADDR> | dhcp enable | tunnel enable }
no wan load-balance nexthop
```

**Параметры**

<ADDR> – IP-адрес назначения (шлюз), задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

dhcp enable – если на интерфейсе IP-адрес получен через DHCP-клиента, используется шлюз с DHCP-сервера.

tunnel enable – использовать в качестве nexthop – peer адрес. Применимо для подключаемых интерфейсов, работающих через PPP.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-VTI  
 CONFIG-LT  
 CONFIG-CELLULAR-MODEM  
 CONFIG-L2TP  
 CONFIG-OPENVPN  
 CONFIG-PPPOE  
 CONFIG-PPTP

### Пример

```
esr(config-gre)# wan load-balance nexthop 16.168.1.25
```

### wan load-balance rule

Данной командой создается правило WAN и осуществляется переход в режим настройки параметров правила.

Использование отрицательной формы команды (no) удаляет созданное WAN-правило.

### Синтаксис

```
wan load-balance rule <ID>
no wan load-balance rule { <ID> | all }
```

### Параметры

<ID> – идентификатор создаваемого правила, принимает значения:

- Для ESR-10/12V/12VF/15/15R/15VF – [1..25];
- Для ESR-20/21/30/31/100/200/1000/1200/1500/1511/1700/3100/3200/3200L/3300 – [1..50].

Значение «all» используется при удалении всех правил WAN для протокола IPv6.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# wan load-balance rule 1
```

### wan load-balance source-address

Данной командой определяется IP-адрес маршрутизатора, который будет использоваться в качестве IP-адреса источника в отправляемых ICMP-пакетах для проверки целей.

Использование отрицательной формы команды (no) удаляет указанный IP-адрес источника.

## Синтаксис

```
wan load-balance source-address <ADDR>  
no wan load-balance source-address
```

## Параметры

<ADDR> – IP-адрес источника, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

## Необходимый уровень привилегий

10

## Командный режим

```
CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-SERIAL  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IF-E1  
CONFIG-IF-MULTILINK  
CONFIG-LT
```

## Пример

```
esr(config)# wan load-balance source-address 16.168.1.25
```

## wan load-balance success-count

Данной командой определяется количество успешных попыток проверки соединения, после которых, в случае успеха, соединение считается вновь активным.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
wan load-balance success-count <VALUE>  
no wan load-balance success-count
```

**Параметры**

<VALUE> – количество попыток, определяется в диапазоне [1...10].

**Значение по умолчанию**

1

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-VTI

CONFIG-LT

CONFIG-CELLULAR-MODEM

CONFIG-L2TP

CONFIG-OPENVPN

CONFIG-PPPOE

CONFIG-PPTP

**Пример**

```
esr(config-if-gi)# wan load-balance success-count 3
```

**wan load-balance target-list**

Данной командой создается список IP-адресов для проверки целостности соединения и осуществляется переход в режим настройки параметров списка.

Использование отрицательной формы команды (no) удаляет созданный список.

**Синтаксис**

```
wan load-balance target-list <NAME>
no wan load-balance target-list { <NAME> | all }
```

**Параметры**

<NAME> – название списка, задается строкой до 31 символа. Значение «all» используется при удалении всех списков IP-адресов для проверки целостности соединения.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# wan load-balance target-list ten1
```

**wan load-balance target-list**

Данной командой производится привязка списка IP-адресов для проверки целостности соединения на сетевом интерфейсе.

Использование отрицательной формы команды (no) удаляет созданный список.

**Синтаксис**

```
wan load-balance target-list <NAME>
no wan load-balance target-list
```

**Параметры**

<NAME> – название списка, задается строкой до 31 символа. Значение «all» используется при удалении всех списков IP-адресов для проверки целостности соединения.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI  
 CONFIG-IF-TE  
 CONFIG-IF-TWE  
 CONFIG-IF-FO  
 CONFIG-IF-HU  
 CONFIG-IF-SUB

CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IF-E1  
CONFIG-IF-MULTILINK  
CONFIG-IP4IP4  
CONFIG-GRE  
CONFIG-VTI  
CONFIG-LT  
CONFIG-CELLULAR-MODEM  
CONFIG-L2TP  
CONFIG-OPENVPN  
CONFIG-PPPOE  
CONFIG-PPTP

### Пример

```
esr(config)# wan load-balance target-list ten1
```

### wan load-balance target-list check-all

Данной командой будут проверяться все IP-адреса из списка проверки целостности. В случае недоступности одного из проверяемых узлов шлюз будет считаться недоступным.

Использование отрицательной формы команды (no) отменяет проверку всех IP-адресов из списка проверки целостности.

### Синтаксис

```
[no] wan load-balance target-list check-all
```

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE  
CONFIG-IF-E1  
CONFIG-IF-MULTILINK  
CONFIG-IP4IP4  
CONFIG-GRE  
CONFIG-VTI  
CONFIG-LT  
CONFIG-CELLULAR-MODEM  
CONFIG-L2TP  
CONFIG-OPENVPN  
CONFIG-PPPOE  
CONFIG-PPTP

### Пример

```
esr(config-gre)# wan load-balance target-list check-all
```

### show wan rules

Данная команда используется для просмотра оперативной информации по правилам WAN для протокола IPv4.

### Синтаксис

```
show wan rules <ID>
```

### Параметры

<ID> – номер правила WAN, принимает значения [1..50].

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
esr# sh wan rules 1
Rule 1 detailed information:
  Failover:      Enabled
  Network: 11.11.11.0/24 Metric: 50
             gi1/0/1 Weight: 2 Nexthop: 3.3.3.1 [Active]
```



## show wan cellular status modem

Данная команда используется для просмотра оперативной информации о состоянии интерфейсов с включенным WAN-режимом для IPv4-стека.

### Синтаксис

```
show wan cellular status modem [ <MODEM-ID> ]
```

### Параметры

<MODEM-ID> – идентификатор ранее созданного 2G/3G/4G-модема (только для IPv4).

В команде можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будут отображены статусы всех интерфейсов заданной группы с включенным WAN-режимом для IPv4-стека. Если задан конкретный интерфейс, то будет отображена детальная информация по данному интерфейсу. При выполнении команды без параметра будут показаны статусы всех системных интерфейсов с включенным WAN-режимом для IPv4-стека.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```

esr-10# show wan cellular status modem 1
Cellular 'modem 1' detailed information:
VRF:                               default
State:                              Active
Nexthop address:                    192.168.8.1 (dhcp)
Last time success (d,h:m:s):        00,00:00:00
Last time failure (d,h:m:s):        00,00:02:24
Uptime (d,h:m:s):                   00,00:02:23
Targets:
  8.8.4.4 Test/Ping State: pass Attempts: 3/3
  8.8.4.4 Test/Ping State: -- Attempts: 0/5

```

## show wan interfaces status

Данная команда используется для просмотра оперативной информации о состоянии интерфейсов с включенным WAN-режимом для IPv4-стека.

### Синтаксис

```
show wan interfaces status [ <IF> ]
```

### Параметры

<IF> – интерфейс, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

В команде можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будут отображены статусы всех интерфейсов заданной группы с включенным WAN-режимом для IPv4-стека. Если задан конкретный интерфейс, то будет отображена детальная информация по данному интерфейсу. При выполнении команды без параметра будут показаны статусы всех системных интерфейсов с включенным WAN-режимом для IPv4-стека.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
esr# show wan interfaces status gigabitethernet 1/0/1
Interface 'gigabitethernet 1/0/1' detailed information:
  State:                Active
  Nexthop address:      3.3.3.1
  Last time success (d,h:m:s): 00,00:00:00
  Last time failure (d,h:m:s): 00,01:48:09
  Uptime (d,h:m:s):    00,01:48:08
  Targets:
    3.3.3.1 Test/Ping State: pass Attempts: 1/1
```

### show wan tunnels status

Данная команда используется для просмотра оперативной информации о состоянии туннелей с включенным WAN-режимом для IPv4-стека.

### Синтаксис

```
show wan tunnels status [ <TUN> ]
```

### Параметры

<INDEX> – идентификатор туннеля, задается в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#).

В команде можно указать несколько туннелей. Если не указывать индексы туннелей, то будут отображены статусы заданной группы всех туннелей с включенным WAN-режимом для IPv4-стека. Если задан конкретный туннель, то будет отображена детальная информация по данному туннелю. При выполнении команды без параметра будут показаны статусы всех туннелей с включенным WAN-режимом для IPv4-стека.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

## Пример

```

esr# show wan tunnels status gre 1
Tunnel 'gre 1' detailed information:
  State:                Active
  Nexthop address:      5.5.5.2
  Last time success (d,h:m:s): 00,00:00:00
  Last time failure (d,h:m:s): 00,00:50:02
  Uptime (d,h:m:s):      00,00:50:01
  Targets:
    5.5.5.2 Test/Ping State: pass Attempts: 1/1

```

## show ipv6 wan rules

Данная команда используется для просмотра оперативной информации по правилам WAN для протокола IPv6.

### Синтаксис

```
show ipv6 wan rules <ID>
```

### Параметры

<ID> – номер правила WAN, принимает значения [1..50].

### Необходимый уровень привилегий

1

### Командный режим

ROOT

## Пример

```

esr# show ipv6 wan rules 1
Rule 1 detailed information:
  Failover:            Enabled
  Network: 2a14::/120 Metric: 0
    bridge 1 Weight: 1 Nexthop: 2a12::1 [Active]

```

## show ipv6 wan interfaces status

Данная команда используется для просмотра оперативной информации о состоянии интерфейсов с включенным WAN-режимом для IPv6-стека.

### Синтаксис

```
show ipv6 wan interfaces status [ <IF> ]
```

## Параметры

<IF> – интерфейс, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

В команде можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будут отображены статусы всех интерфейсов заданной группы интерфейсов с включенным WAN-режимом для стека IPv6. Если задан конкретный интерфейс, то будет отображена детальная информация по данному интерфейсу. При выполнении команды без параметра будут показаны статусы всех системных интерфейсов с включенным WAN-режимом для стека IPv6.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
esr# show ipv6 wan interfaces status bridge 1
Interface 'bridge 1' detailed information:
  State:                Active
  Nexthop address:     2a12::1
  Source address:      fe80::aaf9:4bff:feaa:2dcf
  Last time success (d,h:m:s): 00,00:00:00
  Last time failure (d,h:m:s): 00,00:19:29
  Uptime (d,h:m:s):    00,00:19:28
  Targets:
    2a12::1 Test/Ping State: pass Attempts: 1/1
```

## Настройка объектов отслеживания событий

- [Общие команды настройки Tracking](#)
  - [delay down](#)
  - [delay up](#)
  - [description](#)
  - [enable](#)
  - [mode](#)
  - [show tracks](#)
  - [show track](#)
  - [shutdown track](#)
  - [track](#)
  - [track sla test](#)
  - [track vrrp id](#)
  - [vrrp priority track](#)

### Общие команды настройки Tracking

#### delay down

Данной командой устанавливается задержка перед переходом объекта отслеживания в неактивное состояние.

Использование отрицательной формы команды (no) убирает задержку перед переходом объекта отслеживания в неактивное состояние.

#### Синтаксис

```
delay down <TIME>  
[no] delay down
```

#### Параметры

<TIME> – время задержки в секундах перед переходом объекта отслеживания в неактивное состояние, принимает значение [1..300].

#### Значение по умолчанию

По умолчанию задержка отключена.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-TRACK

#### Пример

```
esr(config-track)# delay down 5
```

## delay up

Данной командой устанавливается задержка перед переходом объекта отслеживания в активное состояние.

Использование отрицательной формы команды (no) убирает задержку перед переходом объекта отслеживания в активное состояние.

### Синтаксис

```
delay up <TIME>  
[no] delay up
```

### Параметры

<TIME> – время задержки в секундах перед переходом объекта отслеживания в неактивное состояние, принимает значение [1..300].

### Значение по умолчанию

По умолчанию задержка отключена.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-TRACK

### Пример

```
esr(config-track)# delay up 5
```

## description

Данной командой определяется описание объекта отслеживания событий.  
Использование отрицательной формы команды (no) удаляет описание.

### Синтаксис

```
description <DESCRIPTION>  
no description
```

### Параметры

<DESCRIPTION> – описание объекта отслеживания событий, задаётся строкой до 255 символов.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-TRACK

### Пример

```
esr(config-track)# description "check virtual router state"
```

### enable

Данной командой включается объект отслеживания.  
Использование отрицательной формы команды (no) отключает объект отслеживания.

### Синтаксис

```
[no] enable
```

### Значение по умолчанию

Объект выключен.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-TRACK

### Пример

```
esr(config-track)# enable
```

### mode

Данной командой задается режим работы объекта отслеживания.  
Использование отрицательной формы команды (no) переводит режим на значение по умолчанию.

### Синтаксис

```
mode <MODE>  
no mode
```

### Параметры

<MODE> – условие нахождения объекта отслеживания в активном состоянии, принимает значения:

- and – объект будет находиться в активном состоянии, если все отслеживаемые условия будут выполняться;
- or – объект будет находиться в активном состоянии, если хотя бы одно отслеживаемое условие будет выполняться.

### Значение по умолчанию

and

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-TRACK

**Пример**

```
esr(config-track)# mode or
```

**show tracks**

Данной командой выводится информация о текущем статусе всех сконфигурированных объектов отслеживания.

**Синтаксис**

```
show tracks
```

**Необходимый уровень привилегий**

10

**Командный режим**

ROOT

**Пример**

```
esr# show tracks
Track ID   State   Last change   Description
-----   -
1          Up      00,00:00:04   DNS probe state
2          Down    00,00:36:04   VRRP-1 Master
```

**show track**

Данной командой выводится информация об объекте отслеживания.

**Синтаксис**

```
show track <TRACK-ID>
```

**Параметры**

<TRACK-ID> – идентификатор объекта отслеживания, принимает значение [1..100].

**Необходимый уровень привилегий**

10



**Командный режим**

ROOT

**Пример**

```

esr# show track 1
Track 1:
  State:          Up
  Changes count: 1 (last 00,00:00:07)
  Mode:           And
  Delay up:       0s
  Delay down:     0s
  Description:    DNS probe state

Conditions:

Type  ID      State  Mode                Last change      VRF
----  -
SLA   1        True   State success      00,00:00:07     --

```

**shutdown track**

Данной командой выполняется управление состоянием интерфейса в зависимости от состояния объекта отслеживания.

Использование отрицательной формы команды (no) отменяет управление состоянием интерфейса.

**Синтаксис**

```

shutdown track <ID>
no shutdown track

```

**Параметры**

<ID> – номер объекта отслеживания, принимает значения в диапазоне [1..100].

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

## CONFIG-LOOPBACK

### Пример

```
esr(config-if-gi)# shutdown track 2
```

### track

Данная команда используется для создания объекта отслеживания (tracking-объекта) и перехода в режим его конфигурирования.

Использование отрицательной формы команды (no) удаляет указанный объект отслеживания.

### Синтаксис

```
[no] track <TRACK-ID>
```

### Параметры

<TRACK-ID> – идентификатор объекта отслеживания, принимает значение [1..100].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# track 10
```

### track sla test

Данной командой устанавливается слежение за состоянием определенного sla-теста.

Использование отрицательной формы команды (no) отключает слежение за состоянием sla-теста.

### Синтаксис

```
track sla test <ID> [mode <MODE> ]  
[no] track sla test <ID>
```

### Параметры

<ID> – идентификатор sla-теста, принимает значение в диапазоне [1..10000].

<MODE> – режим слежения за sla-тестом, может принимать значения:

- state success – отслеживается успешное состояние sla-теста;
- state fail – отслеживается провальное состояние sla-теста;
- reachability – отслеживаются успешность прохождения контрольной фазы с SLA-Responder/ доступность удаленного узла при ispr-тестировании.

### Необходимый уровень привилегий

10

## Командный режим

### CONFIG-TRACK

#### Пример

```
esr(config-track)# track sla test 1 mode state success
```

#### track vrrp id

Данной командой устанавливается слежение за текущим состоянием (ролью) определенного VRRP-маршрутизатора.

При использовании модификатора «not» правило будет срабатывать для всех состояний, за исключением указанного.

Использование отрицательной формы команды (no) отключает слежение за состоянием VRRP-маршрутизатора.

#### Синтаксис

```
track vrrp id <VRID> state [not] <STATE> { master | backup | fault } [ vrf <VRF> ]  
no track vrrp id <VRID> [ vrf <VRF> ]
```

#### Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа;

<VRID> – идентификатор отслеживаемого VRRP-маршрутизатора, принимает значения [1..255];

<STATE> – состояние VRRP-маршрутизатора, при котором объект отслеживания будет менять своё состояние. Может принимать значения:

- master – роль master;
- backup – роль backup;
- fault – fault-состояние VRRP-процесса;

#### Необходимый уровень привилегий

10

## Командный режим

### CONFIG-TRACK

#### Пример

```
esr(config-track)# track vrrp id 2 state master
```

#### vrrp priority track

Данной командой добавляется условие изменения приоритета VRRP-процесса, вступающее в силу при активном состоянии объекта отслеживания.

Использование отрицательной формы команды (no) удаляет условие.

## Синтаксис

```
vrpp priority track <ID> { <PRIO> | increment <INC> | decrement <DEC> }  
no vrpp priority track
```

## Параметры

<ID> – номер объекта отслеживания, принимает значения в диапазоне [1..100];  
<PRIO> – значение приоритета, которое выставится VRRP-процессу при условии активного состояния объекта отслеживания, принимает значения в диапазоне [1..254];  
<INC> – значение, на которое увеличится приоритет VRRP-процесса при условии активного состояния объекта отслеживания, принимает значения в диапазоне [1..254];  
<DEC> – значение, на которое уменьшится приоритет VRRP-процесса при условии активного состояния объекта отслеживания, принимает значения в диапазоне [1..254].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL


CONFIG-BRIDGE

## Пример

```
esr(config-if-gi)# vrpp priority track 2 increment 20
```

## Управление Dual-Homing

- [backup interface](#)
- [backup-interface mac-duplicate](#)
- [backup-interface mac-per-second](#)
- [backup-interface preemption](#)
- [show interfaces backup](#)

 В текущей версии ПО данный функционал поддерживается только на маршрутизаторе ESR-1000.

### backup interface

Данной командой указывается резервный интерфейс, на который будет происходить переключение при потере связи на основном. Включение резервирования возможно только на тех интерфейсах, на которых отключен протокол Spanning Tree и включен VLAN Ingress Filtering.

Использование отрицательной формы команды (no) удаляет настройку с интерфейса.

### Синтаксис

```
backup interface <IF> vlan <VID>
no backup interface
```

### Параметры

<IF> – интерфейс, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

<VID> – идентификационный номер VLAN, задаётся в диапазоне [2...4094]. Можно также задать диапазоном через «-» или перечислением через «,».

### Необходимый уровень привилегий

10

### Командный режим

```
CONFIG-IF-GI
CONFIG-TE
CONFIG-IF-PORT-CHANNEL
```

### Пример

```
esr(config-if-gi)# backup interface gigabitethernet 1/0/15 vlan 10-200
```

### backup-interface mac-duplicate

Данной командой указывается количество копий пакетов с одним и тем же MAC-адресом, которые будут отправлены в активный интерфейс при переключении.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

**Синтаксис**

```
backup-interface mac-duplicate <COUNT>  
no backup-interface mac-duplicate
```

**Параметры**

<COUNT> – количество копий пакетов, принимает значение [1..4].

**Значение по умолчанию**

1

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# backup-interface mac-duplicate 4
```

**backup-interface mac-per-second**

Данной командой указывается количество пакетов в секунду, которое будет отправлено в активный интерфейс при переключении.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

**Синтаксис**

```
backup-interface mac-per-second <COUNT>  
no backup-interface mac-per-second
```

**Параметры**

<COUNT> – количество пакетов в секунду, принимает значение [50..400].

**Значение по умолчанию**

400

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

## Пример

```
esr(config)# backup-interface mac-per-second 200
```

### backup-interface preemption

Данной командой указывается, что необходимо осуществить переключение на основной интерфейс при восстановлении связи. Если настроено восстановление основного интерфейса при активном резервном, то тогда при поднятии линка на основном интерфейсе трафик будет переключен на него.

Использование отрицательной формы команды (no) восстанавливает настройку по умолчанию.

### Синтаксис

```
[no] backup-interface preemption
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Переключение отключено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

## Пример

```
esr(config)# backup-interface preemption
```

### show interfaces backup

Данная команды выводит информацию о состоянии основного и резервного интерфейса.

### Синтаксис

```
show interfaces backup
```

### Необходимый уровень привилегий

1

### Командный режим

ROOT

## Пример

```

esr# show interfaces backup
  Backup Interface Options:
    Preemption is disabled.
    MAC recovery packets rate 400 pps.
    Recovery packets repeats count 1.
VID   Master Interface           Backup Interface           State
----   -
10    gigabitethernet 1/0/3       gigabitethernet 1/0/4       Master Up/Backup Down
----   -
11    gigabitethernet 1/0/3       gigabitethernet 1/0/4       Master Up/Backup Down
----   -
12    gigabitethernet 1/0/3       gigabitethernet 1/0/4       Master Up/Backup Down

```

## Настройка Crypto-Sync

- [crypto-sync](#)
- [enable](#)
- [remote-delete](#)
- [sync crypto force](#)

### crypto-sync

Данной командой осуществляется переход в режим конфигурирования резервирования сертификатов. Использование отрицательной формы команды (no) удаляет настройки резервирования сертификатов.

### Синтаксис

```
[no] crypto-sync
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

## Пример

```

esr(config)# crypto-sync
esr(config-crypto-sync)#

```



**enable**

Данной командой включается функционал для синхронизации сертификатов.  
Использование отрицательной формы команды (no) отключает функционал.

**Синтаксис**

```
[no] enable
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Выключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-CRYPTO-SYNC

**Пример**

```
esr(config-crypto-sync)# enable
```

**remote-delete**

Данной командой включается функция удаления сертификатов при синхронизации.  
Использование отрицательной формы команды (no) выключает функционал.

**Синтаксис**

```
remote-delete
```

```
no remote-delete
```

**Параметры**

Отсутствуют.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-CRYPTO-SYNC

**Пример**

```
esr(config-crypto-sync)# remote-delete
```

**sync crypto force**

Данной командой запускается полная синхронизация файлов в ручном режиме.

**Синтаксис**

```
sync crypto force
```

**Параметры**

Отсутствуют.

**Необходимый уровень привилегий**

10

**Командный режим**

ROOT

**Пример**

```
esr# sync crypto force
```

## Настройка Cluster

- [cluster](#)
- [cluster-interface](#)
- [enable](#)
- [mac-address](#)
- [sync config disable](#)
- [show cluster status](#)
- [show cluster sync status](#)
- [show cluster-unit-licences](#)
- [sync cluster system force](#)
- [unit](#)

### cluster

Данной командой осуществляется переход в режим конфигурирования кластера. Использование отрицательной формы команды (no) удаляет настройки кластера.

### Синтаксис

```
[no] cluster
```

### Параметры

Отсутствуют.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# cluster
esr(config-cluster)#
```

### cluster-interface

Данной командой устанавливается интерфейс, через который будет происходить обмен служебными сообщениями между юнитами в кластере.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
cluster-interface bridge [<BRIDGE-ID>]  
no cluster-interface
```

 В текущей версии в качестве интерфейса можно выбрать только bridge.

## Параметры

<BRIDGE-ID> – идентификационный номер моста, задается в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-CLUSTER

## Пример

```
esr(config-cluster)# cluster-interface bridge 1
```

## enable

Данной командой включается кластер.

Использование отрицательной формы команды (no) выключает кластер.

## Синтаксис

```
[no] enable
```

## Параметры

Отсутствуют.

## Значение по умолчанию

Отключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-CLUSTER

## Пример

```
esr(config-cluster)# enable
```

## mac-address

Данная команда позволяет задать MAC-адрес для определенного юнита.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
mac-address <ADDR>
```

```
no mac-address
```

## Параметры

<ADDR> – MAC-адрес сетевого моста, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF].

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-CLUSTER-UNIT

## Пример

```
esr(config-cluster-unit)# mac-address A8:F9:B0:00:00:04
```

## sync config disable

Данная команда отключает синхронизацию конфигураций в кластере между юнитами.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
[no] sync config disable
```

## Параметры

Отсутствуют.

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-CLUSTER

**Пример**

```
esr(config-cluster)# sync config disable
```

**show cluster status**

Данная команда позволяет посмотреть состояние юнитов в кластере.

**Синтаксис**

```
show cluster status
```

**Необходимый уровень привилегий**

10

**Командный режим**

ROOT

**Пример**

```
esr# show cluster status
Unit      Hostname          Role      MAC address      State      IP address
-----
1*        esr               Active    cc:9d:a2:71:b6:b7  Joined    10.0.0.1
2         esr               Standby   cc:9d:a2:71:b8:ca  Joined    10.0.0.2
```

**show cluster sync status**

Данная команда позволяет посмотреть состояние синхронизации различных подсистем в кластере.

**Синтаксис**

```
show cluster sync status
```

**Необходимый уровень привилегий**

10

**Командный режим**

ROOT

**Пример**

```

esr# show cluster sync status
System part          Synced
-----
candidate-config    Yes
running-config      Yes
SW version           Yes
licence              Yes
licence (After reboot) Yes
date                 Yes

```

**show cluster-unit-licences**

Данная команда позволяет посмотреть загруженные лицензии для устройств в кластере.

**Синтаксис**

```
show cluster-unit-licences
```

**Необходимый уровень привилегий**

10

**Командный режим**

ROOT

**Пример**

```

esr# show cluster-unit-licences
Serial number      Features
-----
NP18000582         BRAS,IPS,WIFI
NP18000622         BRAS,IPS,WIFI

```

**sync cluster system force**

Данная команда вызывает процесс принудительной синхронизации версии ПО, конфигурации, лицензии и времени выбранных юнитов в кластере.

**Синтаксис**

```
sync cluster system force [ unit { <RANGE-ID> | local } ]
```

**Параметры**

<RANGE-ID> – принимает номера юнитов в формате: ID,ID-ID. Номер юнита принимает значения [1..2].

local – подставляет номер локального юнита.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример

```
esr# sync cluster system force
```

### unit

Данной командой осуществляется переход в режим конфигурирования юнита в кластере.

Использование отрицательной формы команды (no) удаляет юнит из кластера.

### Синтаксис

```
[no] unit <ID>
```

### Параметры

<ID> – номер юнита, принимает значения [1..2].

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-CLUSTER

### Пример

```
esr(config-cluster)# unit 1  
esr(config-cluster-unit)#
```



## Настройка общих параметров для failover-сервисов

- [ip failover](#)
- [local-address](#)
- [multicast-address](#)
- [multicast-group](#)
- [remote-address](#)
- [show high-availability state](#)
- [vrrp-group](#)

### ip failover

Данной командой осуществляется переход в режим конфигурирования общих параметров для failover-сервисов.

Использование отрицательной формы команды (no) удаляет настройки общих параметров для failover-сервисов.

### Синтаксис

```
[no] ip failover [ vrf <VRF> ]
```

### Параметры

<VRF> – имя VRF, задается строкой до 31 символа.

### Значение по умолчанию

Отключено

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# ip failover
esr(config-failover)#
```

### local-address

Данной командой устанавливается IP-адрес, на котором failover сервисы принимают failover-сообщения при работе в режиме резервирования.

Использование отрицательной формы команды (no) удаляет установленный IP-адрес.

## Синтаксис

```
local-address { <ADDR> | object-group <NETWORK_OBJ_GROUP_NAME> }  
no local-address
```

## Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

<NETWORK\_OBJ\_GROUP\_NAME> – список адресов, которые будут использоваться в качестве local address.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-FAILOVER

## Пример

```
esr(config-failover)# local-address 192.168.1.1
```

## multicast-address

Данной командой устанавливается многоадресный IP-адрес, который будет использоваться для обмена информации при работе резервирования failover-сервисов в multicast-режиме.

Использование отрицательной формы команды (no) удаляет многоадресный IP-адрес.

## Синтаксис

```
multicast-address <ADDR>  
no multicast-address
```

## Параметры

<ADDR> – многоадресный IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-FAILOVER

## Пример

```
esr(config-failover)# multicast-address 238.0.0.10
```

## multicast-group

Данной командой устанавливается идентификатор multicast-группы для обмена информацией при работе резервирования failover-сервисов в multicast-режиме.

Использование отрицательной формы команды (no) удаляет идентификатор группы.

### Синтаксис

```
multicast-group <GROUP>
no multicast-group
```

### Параметры

<GROUP> – multicast-группа, указывается в диапазоне [1000..9999].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-FAILOVER

### Пример

```
esr(config-failover)# multicast-group 1028
```

## remote-address

Данной командой устанавливается IP-адрес, на который failover-сервисы отправляют failover-сообщения при работе в режиме резервирования.

Использование отрицательной формы команды (no) удаляет установленный IP-адрес.

### Синтаксис

```
remote-address { <ADDR> | object-group <NETWORK_OBJ_GROUP_NAME> }
no remote-address
```

### Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<NETWORK\_OBJ\_GROUP\_NAME> – список адресов, которые будут использоваться в качестве local address.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-FAILOVER

## Пример

```
esr(config-failover)# remote-address 192.168.1.2
```

## show high-availability state

Данная команда позволяет посмотреть общее состояние систем резервирования и роль устройства.

## Синтаксис

```
show high-availability state
```

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
wlc-30r# show high-availability state
VRRP role:                               Master
AP Tunnels:
  State:                                  Successful synchronization
  Last synchronization:                   2024-11-25 15:39:10
DHCP option 82 table:
  State:                                  Disabled
  Last state change:                       --
DHCP server:
VRF:
  State:                                  Successful synchronization
  Last synchronization:                   2024-11-25 15:39:23
crypto-sync:
  State:                                  Successful synchronization
  Last synchronization:                   2024-11-25 15:39:24
Firewall:
  State:                                  Disabled
  Last state change:                       --
WLC:
  State:                                  Successful synchronization
  Last synchronization:                   2024-11-25 15:39:24
WEB profiles:
  State:                                  Disabled
```

## vrrp-group

Данной командой указывается VRRP-группа, по состоянию которой будет определяться мастерство при работе failover сервисов в режиме Active-Standby.

Использование отрицательной формы команды (no) удаляет указанную vrrp-group из конфигурации failover.

### Синтаксис

```
vrrp-group <GRID>
```

```
no vrrp-group
```

### Параметры

<GRID> – идентификатор группы VRRP-маршрутизатора, принимает значения [1..32].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-FAILOVER

### Пример

```
esr(config-failover)# vrrp-group 2
```

## 26 Безопасность

- Управление списками контроля доступа (ACL)
- Управление Firewall
- Управление логированием и защитой от сетевых атак
- Управление фильтрацией
- Управление системой предотвращения вторжений (IPS/IDS)

### Управление списками контроля доступа (ACL)

- action
- description
- enable
- ip access-list extended
- match cos
- match destination-address
- match destination-mac
- match destination-port
- match dscp
- match ip-precedence
- match protocol
- match source-address
- match source-mac
- match source-port
- match vlan
- rule
- service-acl input
- show ip access-list

#### action

Данная команда используется для указания действия, которое должно быть применено для трафика, удовлетворяющего заданным критериям.

Использование отрицательной формы команды (no) удаляет назначенное действие.

#### Синтаксис

```
action <ACT>
```

```
no action
```

#### Параметры

<ACT> – назначаемое действие:

- permit – прохождение трафика разрешается;
- deny – прохождение трафика запрещается.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-ACL-RULE

## Пример

```
esr(config-acl-rule)# action permit
```

## description

Данная команда используется для изменения описания конфигурируемого списка контроля доступа. Использование отрицательной формы команды (no) удаляет установленное описание.

## Синтаксис

```
description <DESCRIPTION>  
no description
```

## Параметры

<DESCRIPTION> – описание списка контроля доступа, задаётся строкой до 255 символов.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ACL

## Пример

```
esr(config-acl)# description "Drop SSH traffic"
```

## enable

Данная команда используется для активирования правила.

Использование отрицательной формы команды (no) деактивирует правило.

## Синтаксис

```
[no] enable
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Правило выключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ACL-RULE

### Пример

```
esr(config-acl-rule)# enable
```

## ip access-list extended

Данная команда используется для создания списка контроля доступа и перехода в режим конфигурирования списка.

Использование отрицательной формы команды (no) удаляет заданный список контроля доступа.

### Синтаксис

```
[no] ip access-list extended <NAME>
```

### Параметры

<NAME> – имя создаваемого списка контроля доступа, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG

### Пример

```
esr(config)# ip access-list extended acl-ssh-drop  
esr(config-acl)#
```

## match cos

Данной командой устанавливается значение 802.1p приоритета, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
match cos <COS>
```

```
no match cos
```

### Параметры

<COS> – значение 802.1p приоритета, принимает значения [0..7].



## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ACL-RULE

## Пример

```
esr(config-acl-rule)# match cos 2
```

## match destination-address

Данной командой устанавливаются IP-адреса получателя, для которых должно срабатывать правило. Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
match destination-address { <ADDR> <MASK> | any }  
no match destination-address
```

## Параметры

<ADDR> – IP-адрес получателя, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<MASK> – маска IP-адреса, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Биты маски, установленные в 0, задают биты IP-адреса, исключаемые из сравнения при поиске. При указании значения «any» правило будет срабатывать для любого IP-адреса получателя.

## Значение по умолчанию

any

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ACL-RULE

## Пример

```
esr(config-acl-rule)# match destination-address 10.10.10.0 255.255.255.0
```

## match destination-mac

Данной командой устанавливаются MAC-адреса получателя, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
match destination-mac <ADDR> <WILDCARD>
```

```
no match destination-mac
```

### Параметры

<ADDR> – MAC-адрес получателя, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF];

<WILDCARD> – маска MAC-адреса, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF]. Биты маски, установленные в 0, задают биты MAC-адреса, исключаемые из сравнения при поиске.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-ACL-RULE

### Пример

```
esr(config-acl-rule)# match destination-mac A8:F9:4B:AA:00:41 00:00:00:00:00:FF
```

### match destination-port

Данной командой устанавливается номер TCP/UDP-порта получателя, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) удаляет назначение.

### Синтаксис

```
match destination-port <TYPE> {<PORT> | <PORT>-<PORT>}
```

```
no match destination-port
```

### Параметры

<TYPE> – принимает значения «any» или «port-range»;

Можно указать один порт либо указать диапазон портов через «-»;

<PORT> – продолжение команды port-range, номер TCP/UDP-порта отправителя, принимает значения [1..65535]. При указании значения «any» правило будет срабатывать для любого TCP/UDP-порта отправителя.

### Значение по умолчанию

any

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ACL-RULE

## Пример

```
esr(config-acl-rule)# match destination-port port-range 22
```

## match dscp

Данной командой устанавливается значение кода DSCP, для которого должно срабатывать правило. Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
match dscp <DSCP>
```

```
no match dscp
```

## Параметры

<DSCP> – значение кода DSCP, принимает значения [0..63].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ACL-RULE

## Пример

```
esr(config-acl-rule)# match dscp 55
```

## match ip-precedence

Данной командой устанавливается значение кода IP Precedence, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
match ip-precedence <IPP>
```

```
no match ip-precedence
```

## Параметры

<IPP> – значение кода IP Precedence, принимает значения [0..7].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ACL-RULE

## Пример

```
esr(config-acl-rule)# match ip-precedence 5
```

## match protocol

Данной командой устанавливается имя IP-протокола, для которого должно срабатывать правило. Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
match protocol <TYPE>  
no match protocol  
match protocol-id <ID>  
no match protocol-id
```

## Параметры

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre. При указании значения «any» правило будет срабатывать для любых протоколов;

<ID> – идентификационный номер IP-протокола, принимает значения [0x00-0xFF].

При указании значения «any» правило будет срабатывать для любого протокола.

## Значение по умолчанию

any

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ACL-RULE

## Пример

```
esr(config-acl-rule)# match protocol tcp
```

### match source-address

Данной командой устанавливаются IP-адреса отправителя, для которых должно срабатывать правило. Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
match source-address { <ADDR> <MASK> | any }  
no match source-address
```

### Параметры

<ADDR> – IP-адрес отправителя, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<MASK> – маска IP-адреса, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Биты маски, установленные в 0, задают биты IP-адреса, исключаемые из сравнения при поиске. При указании значения «any» правило будет срабатывать для любого IP-адреса отправителя.

### Значение по умолчанию

any

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-ACL-RULE

## Пример

```
esr(config-acl-rule)# match source-address 10.100.100.0 255.255.255.0
```

### match source-mac

Данной командой устанавливаются MAC-адреса отправителя, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
match source-mac <ADDR> <WILDCARD>  
no match source-mac
```

## Параметры

<ADDR> – MAC-адрес отправителя, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF];

<WILDCARD> – маска MAC-адреса, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF]. Биты маски, установленные в 0, задают биты MAC-адреса, исключаемые из сравнения при поиске.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ACL-RULE

## Пример

```
esr(config-acl-rule)# match source-mac A8:F9:4B:AA:00:40 00:00:00:FF:FF:FF
```

## match source-port

Данной командой устанавливается номер TCP/UDP-порта отправителя, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
match source-port <TYPE> { <PORT> | <PORT>-<PORT> }
```

```
no match source-port
```

## Параметры

<TYPE> – принимает значения «any» или «port-range»;

Можно указать один порт либо указать диапазон портов через «-»;

<PORT> – Продолжение команды port-range, номер TCP/UDP-порта отправителя, принимает значения [1..65535]. При указании значения «any» правило будет срабатывать для любого TCP/UDP-порта отправителя.

## Значение по умолчанию

any

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ACL-RULE

## Пример

```
esr(config-acl-rule)# match source-port any
```

## match vlan

Данной командой устанавливается значение идентификационного номера VLAN, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
match vlan <VID>
```

```
no match vlan
```

## Параметры

<VID> – идентификационный номер VLAN, принимает значения [1...4094].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ACL-RULE

## Пример

```
esr(config-acl-rule)# match vlan 100
```

## rule

Данная команда используется для создания правила и перехода в режим конфигурирования CONFIG-ACL-RULE. Правила обрабатываются устройством в порядке возрастания их номеров.

Использование отрицательной формы команды (no) удаляет указанное правило.

## Синтаксис

```
[no] rule <ORDER>
```

## Параметры

<ORDER> – номер правила, принимает значения [1..4094].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ACL


### Пример

```
esr(config-acl)# rule 10  
esr(config-acl-rule)#
```

### service-acl input

Данная команда используется для привязки указанного списка контроля доступа к конфигурируемому интерфейсу для фильтрации входящего трафика.

Использование отрицательной формы команды (no) удаляет привязку списка контроля доступа к данному интерфейсу.

 Использование фильтрации пакетов при помощи ACL на физическом интерфейсе увеличивает нагрузку на CPU маршрутизатора.  
Для фильтрации трафика рекомендуется использовать функционал ZoneBased Firewall.

### Синтаксис

```
service-acl input <NAME>
```

```
no service-acl input
```

### Параметры

<NAME> – имя списка контроля доступа, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

15

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

### Пример

```
esr(config-if-gi)# service-acl input acl-ssh-drop
```



## show ip access-list

Данная команда используется для просмотра списков управления доступом.

### Синтаксис

```
show ip access-list [ <NAME> [ <ORDER> ] ]
```

### Параметры

<NAME> – имя списка управления доступом, задаётся строкой до 31 символа;

<ORDER> – номер правила, принимает значения [1..4096]. При указании номера правила будет показана информация только по данному правилу.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```

esr# show ip access-list
Name                               Description
-----
acl-telnet-drop                    --
acl-ssh-drop                       Drop SSH traffic
esr# show ip access-list acl-ssh-drop
Index:                              1
Matching pattern:
  Protocol:                          TCP(6)
  Source MAC address:                 any
  Source IP address:                  any
  Source port:                         any
  Destination MAC address:            any
  Destination IP address:             any
  Destination port:                   22
Action:                               Deny
Status:                               Enabled
-----
Index:                              2
Matching pattern:
  Protocol:                            any
  Source MAC address:                  any
  Source IP address:                   any
  Destination MAC address:             any
  Destination IP address:              any
Action:                               Permit
Status:                               Enabled
-----

```

## Управление Firewall

- [action](#)
- [check output-interface](#)
- [clear ip firewall counters](#)
- [clear ip firewall sessions](#)
- [clear ipv6 firewall counters](#)
- [clear ipv6 firewall sessions](#)
- [description](#)
- [enable](#)
- [ip firewall disable](#)
- [ip firewall mode](#)
- [ip firewall sessions counters](#)
- [ip firewall sessions allow-unknown](#)
- [ip firewall sessions generic-timeout](#)
- [ip firewall sessions icmp-timeout](#)
- [ip firewall sessions icmpv6-timeout](#)
- [ip firewall sessions max-expect](#)
- [ip firewall sessions max-tracking](#)
- [ip firewall sessions tcp-connect-timeout](#)
- [ip firewall sessions tcp-disconnect-timeout](#)
- [ip firewall sessions tcp-established-timeout](#)
- [ip firewall sessions tcp-latecome-timeout](#)
- [ip firewall sessions tracking](#)
- [ip firewall sessions udp-assured-timeout](#)
- [ip firewall sessions udp-wait-timeout](#)
- [match application](#)
- [match destination-address](#)
- [match destination-address-port](#)
- [match destination-mac](#)
- [match destination-nat](#)
- [match destination-port](#)
- [match fragment](#)
- [match icmp](#)
- [match ip-option](#)
- [match protocol](#)
- [match source-address](#)
- [match source-address-port](#)
- [match source-mac](#)
- [match source-port](#)
- [ports firewall enable](#)
- [rearrange](#)
- [renumber rule](#)
- [rule](#)
- [security zone](#)
- [security-zone](#)
- [security zone-pair](#)
- [show ip firewall counters](#)
- [show ip firewall sessions](#)
- [show ip firewall sessions tracking](#)
- [show ipv6 firewall counters](#)
- [show ipv6 firewall sessions](#)
- [show security zone](#)
- [show security zone-pair](#)
- [show security zone-pair configuration](#)

## action

Данная команда используется для указания действия, которое должно быть применено для трафика, удовлетворяющего заданным критериям.

Использование отрицательной формы команды (no) удаляет назначенное действие.


## Синтаксис

```
action { permit | deny | reject | netflow-sample | sflow-sample | rate-limit total pps
<RATE> | session-limit [total <SESSION>] [per-source-host <SESSION>] } [log]
```

```
no action
```

## Параметры

- permit – прохождение трафика разрешается;
- deny – прохождение трафика запрещается;
- reject – прохождение трафика запрещается, а также посылается отправителю ответ об ошибке;
- netflow-sample – прохождение трафика разрешается, осуществляется экспорт статистики по протоколу Netflow;
- sflow-sample – прохождение трафика разрешается, осуществляется экспорт статистики по протоколу sFlow;
- rate-limit total pps <RATE> – прохождение трафика разрешается, ограничивается количество обрабатываемых правилом пакетов в секунду. Команда применима только в правилах между зонами any self:
  - <RATE> – количество пакетов в секунду, принимает значения [1..10000].
- session-limit [total <SESSION>] [per-source-host <SESSION>] – прохождение трафика разрешается и ограничивается число одновременных сессий трафика:
  - total <SESSION> – ограничивается общее число сессий трафика, попадающих под данное правило:
    - <SESSION> – число одновременных сессий, принимает значения [1..10000].
  - [per-source-host <SESSION>] – ограничивается число сессий трафика от одного хоста источника:
    - <SESSION> – число одновременных сессий, принимает значения [1..10000].
- log – ключ для активации логирования сессий, устанавливаемым согласно данному правилу.

 Функционал session-limit доступен только на моделях ESR-30, ESR-31, ESR-3100, ESR-3200, ESR-3200L, ESR-3300.

## Значение по умолчанию

Действие не настроено, логирование отключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-SECURITY-ZONE-PAIR-RULE

## Пример

```
esr(config-security-zone-pair-rule)# action permit
```

## check output-interface

Данной командой разрешается очистка Firewall-сессий, в случае если выходной интерфейс изменен. Использование отрицательной формы команды (no) удаляет назначенное действие.

## Синтаксис

```
check output-interface
no check output-interface
```

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-SECURITY-ZONE-PAIR-RULE

## Пример

```
esr(config-security-zone-pair-rule)# check output-interface
```

## clear ip firewall counters

Данной командой осуществляется сброс счетчиков правил Firewall.

## Синтаксис

```
clear ip firewall counters [ vrf <VRF> ] [ <SOURCE-ZONE> [ <DESTINATION-ZONE>
[<ORDER>] ] ]
```

## Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут очищены счетчики правил в указанном VRF;

<SOURCE-ZONE> – зона безопасности, из которой поступает трафик;

<DESTINATION-ZONE> – зона безопасности, в которую поступает трафик;

<ORDER> – номер правила, принимает значения [1..10000]. При указании номера правила будут очищены счетчики только по данному правилу.

## Необходимый уровень привилегий

10

**Командный режим**

ROOT

**Пример**

```
esr# clear ip firewall counters trusted self
```

**clear ip firewall sessions**

Данной командой осуществляется удаление активных IP-сессий.

**Синтаксис**

```
clear ip firewall sessions [ vrf <VRF> ] [ protocol <TYPE> ] [ inside-source-address
<ADDR> ] [ outside-source-address <ADDR> ] [ inside-destination-address <ADDR> ]
[ outside-destination-address <ADDR> ] [ inside-source-port <PORT> ] [ outside-source-
port <PORT> ] [ inside-destination-port <PORT> ] [ outside-destination-port <PORT> ]
```

**Параметры**

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут удалены активные сессии в указанном VRF;

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<PORT> – TCP/UDP-порт, принимает значения [1..65535];

inside-source-address – ключ для указания IP-адреса источника входящих пакетов;

inside-destination-address – ключ для указания IP-адреса назначения входящих пакетов;

outside-source-address – ключ для указания IP-адреса источника отправляемых пакетов;

outside-destination-address – ключ для указания IP-адреса назначения отправляемых пакетов;

inside-source-port – ключ для указания TCP/UDP-порта отправителя в входящих пакетах;

outside-source-port – ключ для указания TCP/UDP-порта отправителя в отправляемых пакетах;

inside-destination-port – ключ для указания TCP/UDP-порта назначения в входящих пакетах;

outside-destination-port – ключ для указания TCP/UDP-порта назначения в отправляемых пакетах.

**Необходимый уровень привилегий**

10

**Командный режим**

ROOT

**Пример**

```
esr# clear ip firewall sessions vrf VRF1
```

## clear ipv6 firewall counters

Данной командой осуществляется сброс счетчиков правил Firewall.

### Синтаксис

```
clear ipv6 firewall counters [ vrf <VRF> ] [ <SOURCE-ZONE> [ <DESTINATION-ZONE>
[<ORDER>] ] ]
```

### Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут очищены счетчики правил в указанном VRF;

<SOURCE-ZONE> – зона безопасности, из которой поступает трафик;

<DESTINATION-ZONE> – зона безопасности, в которую поступает трафик;

<ORDER> – номер правила, принимает значения [1..10000]. При указании номера правила будут очищены счетчики только по данному правилу.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример

```
esr# clear ipv6 firewall counters trusted self
```

## clear ipv6 firewall sessions

Данной командой осуществляется удаление активных IPv6-сессий.

### Синтаксис

```
clear ipv6 firewall sessions [ vrf <VRF> ] [ protocol <TYPE> ] [ inside-source-address
<IPV6-ADDR> ] [ outside-source-address <IPV6-ADDR> ] [ inside-destination-address <IPV6-
ADDR> ] [ outside-destination-address <IPV6-ADDR> ] [ inside-source-port <PORT> ]
[ outside-source-port <PORT> ] [ inside-destination-port <PORT> ] [ outside-destination-
port <PORT> ]
```

### Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут удалены активные сессии в указанном VRF;

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;

<IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<PORT> – TCP/UDP-порт, принимает значения [1..65535];

inside-source-address – команда для указания IPv6-адреса источника входящих пакетов;  
 inside-destination-address – команда для указания IPv6-адреса назначения входящих пакетов;  
 outside-source-address – команда для указания IPv6-адреса источника отправляемых пакетов;  
 outside-destination-address – команда для указания IPv6-адреса назначения отправляемых пакетов;  
 inside-source-port – ключ для указания TCP/UDP-порта отправителя в входящих пакетах;  
 outside-source-port – ключ для указания TCP/UDP-порта отправителя в отправляемых пакетах;  
 inside-destination-port – ключ для указания TCP/UDP-порта назначения в входящих пакетах;  
 outside-destination-port – ключ для указания TCP/UDP-порта назначения в отправляемых пакетах.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример

```
esr# clear ipv6 firewall sessions vrf VRF1
```

### description

Данная команда используется для изменения описания конфигурируемой зоны или пары зон безопасности. Использование отрицательной формы команды (no) удаляет установленное описание.

### Синтаксис

```
description <DESCRIPTION>  
no description
```

### Параметры

<DESCRIPTION> – описание зоны безопасности, задаётся строкой до 255 символов.

### Необходимый уровень привилегий

10

### Командный режим

```
CONFIG-SECURITY-ZONE  
CONFIG-SECURITY-ZONE-PAIR  
CONFIG-SECURITY-ZONE-PAIR-RULE
```

### Пример

```
esr(config-security-zone)# description "Trusted interfaces"
```

**enable**

Данная команда используется для активирования правила.

Использование отрицательной формы команды (no) деактивирует правило.

**Синтаксис**

```
[no] enable
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-SECURITY-ZONE-PAIR-RULE

**Пример**

```
esr(config-security-zone-rule)# enable
```

**ip firewall disable**

Данная команда используется для отключения функции Firewall на сетевом интерфейсе.

Использование отрицательной формы команды (no) включает функцию Firewall на сетевом интерфейсе.

**Синтаксис**

```
[no] ip firewall disable
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE



CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-OOB  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-SERIAL  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IF-E1  
CONFIG-IF-MULTILINK  
CONFIG-CELLULAR-MODEM  
CONFIG-VTI  
CONFIG-GRE  
CONFIG-IP4IP4  
CONFIG-L2TP  
CONFIG-LT  
CONFIG-PPPOE  
CONFIG-PPTP  
CONFIG-OPENVPN  
CONFIG-WIREGUARD-SERVER  
CONFIG-WIREGUARD-TUNNEL-PEER

### Пример

```
esr(config-if-gi)# ip firewall disable
```

### ip firewall mode

Данная команда используется для выбора режима работы межсетевого экрана.

Использование отрицательной формы команды (no) устанавливает режим работы межсетевого экрана по умолчанию.

### Синтаксис

```
ip firewall mode <MODE>  
no ip firewall mode
```

### Параметры

<MODE> – режим работы межсетевого экрана, может принимать значения:

- **stateful** – режим, при котором маршрутизатор отслеживает сессии. Первые пакеты сессии проходят полный цикл проверки согласно правилам межсетевого экрана, а последующие пакеты сессии маршрутизируются без дополнительных проверок. В этом режиме, если "прямой" трафик

разрешён, "ответный" трафик разрешается автоматически. Данное правило не распространяется на работу механизма DPI.

- `stateless` – режим, при котором маршрутизатор не отслеживает сессии. Каждый пакет проходит полный цикл проверки согласно правилам межсетевого экрана, что существенно снижает производительность оборудования. Использование данного режима допустимо только в условиях крайней необходимости.

### Значение по умолчанию

`stateful`

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

### Пример

```
esr(config)# ip firewall mode stateless
```

### `ip firewall sessions counters`

Командой выполняется включение счетчиков сессий для NAT и Firewall. Счетчики увеличиваются только тогда, когда устанавливается новая сессия. Для установленных сессий увеличения значений счетчиков не происходит при прохождении пакетов. Включение счетчиков снижает производительность маршрутизатора.

Команды для просмотра счетчиков и сессий описаны в разделах [show ip firewall counters](#), [show ip firewall sessions](#), [show ipv6 firewall counters](#) и [show ipv6 firewall sessions](#).

Использование отрицательной формы команды (`no`) отключает счетчики сессий.

### Синтаксис

```
[no] ip firewall sessions counters
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

## Пример

```
esr(config)# ip firewall sessions counters
```

### **ip firewall sessions allow-unknown**

Данной командой отключается фильтрация пакетов, для которых не удалось определить принадлежность к какому-либо известному соединению и которые не являются началом нового соединения.

Использование отрицательной формы команды (no) включает фильтрацию.

### **Синтаксис**

```
[no] ip firewall sessions allow-unknown
```

### **Параметры**

Команда не содержит параметров.

### **Значение по умолчанию**

Включено.

### **Необходимый уровень привилегий**

15

### **Командный режим**

CONFIG

## Пример

```
esr(config)# ip firewall sessions allow-unknown
```

### **ip firewall sessions generic-timeout**

Данной командой определяется время жизни сессии для неподдерживаемых протоколов, по истечении которого она считается устаревшей и удаляется из таблицы отслеживаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### **Синтаксис**

```
ip firewall sessions generic-timeout <TIME>
```

```
no ip firewall sessions generic-timeout
```

### **Параметры**

<TIME> – время жизни сессии для неподдерживаемых протоколов, принимает значения в секундах [1..8553600].

**Значение по умолчанию**

60 секунд.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip firewall sessions generic-timeout 60
```

**ip firewall sessions icmp-timeout**

Данной командой определяется время жизни ICMP-сессии, по истечении которого она считается устаревшей и удаляется из таблицы отслеживаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
ip firewall sessions icmp-timeout <TIME>
```

```
no ip firewall sessions icmp-timeout
```

**Параметры**

<TIME> – время жизни ICMP-сессии, принимает значения в секундах [1..8553600].

**Значение по умолчанию**

30 секунд.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip firewall sessions icmp-timeout 60
```

## ip firewall sessions icmpv6-timeout

Данной командой определяется время жизни ICMPv6-сессии, по истечении которого она считается устаревшей и удаляется из таблицы отслеживаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ip firewall sessions icmpv6-timeout <TIME>  
no ip firewall sessions icmpv6-timeout
```

### Параметры

<TIME> – время жизни ICMPv6-сессии, принимает значения в секундах [1..8553600].

### Значение по умолчанию

30 секунд.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

### Пример

```
esr(config)# ip firewall sessions icmpv6-timeout 60
```

## ip firewall sessions max-expect

Данной командой определяется размер таблицы сессий, ожидающих обработки.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ip firewall sessions max-expect <COUNT>  
no ip firewall sessions max-expect
```

### Параметры

<COUNT> – размер таблицы, принимает следующие значения:

- ESR-1511, ESR-1700, ESR-3100, ESR-3200L, ESR-3200, ESR-3300, vESR – 8553600
- ESR-1500 – 2430000
- ESR-1000, ESR-1200 – 3130000
- ESR-200 – 2259000
- ESR-100 – 1574000

- ESR-3x – 3260000
- ESR-2x – 2940000
- ESR-15 – 300000
- ESR-1x – 440000

**Значение по умолчанию**

256

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip firewall sessions max-expect 512
```

**ip firewall sessions max-tracking**

Данной командой определяется размер таблицы отслеживаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
ip firewall sessions max-tracking <COUNT>
```

```
no ip firewall sessions max-tracking
```

**Параметры**

<COUNT> – размер таблицы, принимает следующие значения:

- ESR-1511, ESR-1700, ESR-3100, ESR-3200L, ESR-3200, ESR-3300, vESR – 8553600
- ESR-1500 – 2430000
- ESR-1000, ESR-1200 – 3130000
- ESR-200 – 2259000
- ESR-100 – 1574000
- ESR-3x – 3260000
- ESR-2x – 2940000
- ESR-15 – 300000
- ESR-1x – 440000

**Значение по умолчанию**

ESR-1x, ESR-15 – 64000

Для остальных моделей – 512000

**Необходимый уровень привилегий**

15

## Командный режим

CONFIG

### Пример

```
esr(config)# ip firewall sessions max-tracking 256000
```

### ip firewall sessions tcp-connect-timeout

Данной командой определяется время жизни TCP-сессии в состоянии «соединение устанавливается», по истечении которого она считается устаревшей и удаляется из таблицы отслеживаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ip firewall sessions tcp-connect-timeout <TIME>
```

```
no ip firewall sessions tcp-connect-timeout
```

### Параметры

<TIME> – время жизни TCP-сессии в состоянии «соединение устанавливается», принимает значения в секундах [1..8553600].

### Значение по умолчанию

60 секунд.

### Необходимый уровень привилегий

15

## Командный режим

CONFIG

### Пример

```
esr(config)# ip firewall sessions tcp-connect-timeout 120
```

### ip firewall sessions tcp-disconnect-timeout

Данной командой определяется время жизни TCP-сессии в состоянии «соединение закрывается», по истечении которого она считается устаревшей и удаляется из таблицы отслеживаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ip firewall sessions tcp-disconnect-timeout <TIME>
```

```
no ip firewall sessions tcp-disconnect-timeout
```

## Параметры

<TIME> – время жизни TCP-сессии в состоянии «соединение закрывается», принимает значения в секундах [1..8553600].

## Значение по умолчанию

30 секунд

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# ip firewall sessions tcp-disconnect-timeout 10
```

## ip firewall sessions tcp-established-timeout

Данной командой определяется время жизни TCP-сессии в состоянии «соединение установлено», по истечении которого она считается устаревшей и удаляется из таблицы отслеживаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
ip firewall sessions tcp-established-timeout <TIME>
```

```
no ip firewall sessions tcp-established-timeout
```

## Параметры

<TIME> – время жизни TCP-сессии в состоянии «соединение установлено», принимает значения в секундах [1..8553600].

## Значение по умолчанию

120 секунд.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG



## Пример

```
esr(config)# ip firewall sessions tcp-established-timeout 3600
```

## ip firewall sessions tcp-latecome-timeout

Данной командой определяется время ожидания, по истечении которого происходит фактическое удаление закрытой TCP-сессии из таблицы отслеживаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
ip firewall sessions tcp-latecome-timeout <TIME>
```

```
no ip firewall sessions tcp-latecome-timeout
```

## Параметры

<TIME> – время ожидания, принимает значения в секундах [1..8553600].

## Значение по умолчанию

120 секунд.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# ip firewall sessions tcp-latecome-timeout 10
```

## ip firewall sessions tracking

Данной командой включается функция отслеживания сессий уровня приложений для отдельных протоколов.

Использование отрицательной формы команды (no) отключает функцию отслеживания сессий уровня приложений для отдельных протоколов.

## Синтаксис

```
ip firewall sessions tracking { <PROTOCOL> | sip [ port <OBJECT-GROUP-SERVICE> | session-lifetime <TIME> ] }
```

```
no ip firewall sessions tracking { <PROTOCOL> | sip [ port <OBJECT-GROUP-SERVICE> ] | all }
```

## Параметры

<PROTOCOL> – протокол уровня приложений, сессии которого должны отслеживаться, принимает значения [ftp, h323, pptp, netbios-ns];

<OBJECT-GROUP-SERVICE> – имя профиля TCP/UDP-портов SIP-сессии, задаётся строкой до 31 символа. Если группа не указана, то отслеживание сессий SIP будет осуществляться для порта 5060;

Вместо имени отдельного протокола можно использовать ключ "all", который включает функцию отслеживания сессий уровня приложений для всех доступных протоколов.

<TIME> – время жизни отслеживаемой SIP-сессии, по истечении которого она считается устаревшей и удаляется из таблицы отслеживаемых сессий. Принимает значения в секундах [1..8553600].

## Значение по умолчанию

Отключено для всех протоколов.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# ip firewall sessions tracking ftp
```

## ip firewall sessions udp-assured-timeout

Данной командой определяется время жизни UDP-сессии в состоянии «соединение подтверждено», по истечении которого она считается устаревшей и удаляется из таблицы отслеживаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
ip firewall sessions udp-assured-timeout <TIME>
```

```
no ip firewall sessions udp-assured-timeout
```

## Параметры

<TIME> – время жизни UDP-сессии в состоянии «соединение подтверждено», принимает значения в секундах [1..8553600].

## Значение по умолчанию

180 секунд.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

### Пример

```
esr(config)# ip firewall sessions udp-assured-timeout 3600
```

### ip firewall sessions udp-wait-timeout

Данной командой определяется время жизни UDP-сессии в состоянии «соединение не подтверждено», по истечении которого она считается устаревшей и удаляется из таблицы отслеживаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ip firewall sessions udp-wait-timeout <TIME>
```

```
no ip firewall sessions udp-wait-timeout
```

### Параметры

<TIME> – время жизни UDP-сессии в состоянии «соединение не подтверждено», принимает значения в секундах [1..8553600].

### Значение по умолчанию

30 секунд.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG

### Пример

```
esr(config)# ip firewall sessions udp-wait-timeout 60
```

### match application

Данной командой устанавливается профиль приложений, для которых должно срабатывать правило. Данная функция используется для фильтрации по приложениям (механизм DPI).

При использовании параметра «not» правило будет срабатывать для приложений, которые не входят в указанный профиль.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
match [not] application <OBJ-GROUP-APPLICATION>
no match application
```

## Параметры

<OBJ-GROUP-APPLICATION> – имя профиля приложений, задаётся строкой до 31 символа.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-SECURITY-ZONE-PAIR-RULE

## Пример

```
esr(config-security-zone-pair-rule)# match application APP_DENY
```

## match destination-address

Данной командой устанавливается профиль IP-адресов получателя, для которых должно срабатывать правило.

При использовании параметра «not» правило будет срабатывать для IP-адресов получателя, которые не входят в указанный профиль.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
match [not] destination-address { address-range { <ADDR>[-<ADDR>] |
<IPV6-ADDR>[-<IPV6-ADDR>] } | prefix { <ADDR/LEN> | <IPv6-ADDR/LEN> } | object-group
<OBJ-GROUP-NETWORK-NAME> | any }
no match destination-address
```

## Параметры

address-range <ADDR>[-<ADDR>] – диапазон IP-адресов для правил firewall. Если не указывать IP-адрес конца диапазона, то в качестве IP-адреса для срабатывания правила используется только IP-адрес начала диапазона.

Параметр задаётся в виде A.B.C.D, где каждая часть принимает значения [0..255]; <IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

prefix <ADDR/LEN> – IP-подсеть, используемая для срабатывания правила фильтрации firewall.

Параметр задаётся в виде A.B.C.D/E, где каждая часть A – D принимает значения [0..255] и E принимает значения [1..32]; <IPv6-ADDR/LEN> – IPv6-адрес, задаётся в виде X:X:X::X/E, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и E принимает значения [1..128];

object-group <OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа.

При указании значения «any» правило будет срабатывать для любого IP-адреса получателя.

**Значение по умолчанию**

any

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-SECURITY-ZONE-PAIR-RULE

**Пример**

```
esr(config-security-pair-zone-rule)# match destination-address object-group remote_workspace
```

**match destination-address-port**

Данной командой устанавливается профиль связок IP-адресов и TCP/UDP-портов получателя, для которых должно срабатывать правило.

При использовании параметра «not» правило будет срабатывать для связок IP-адресов и TCP/UDP-портов получателя, которые не входят в указанный профиль.

Использование отрицательной формы команды (no) отменяет назначение.

**Синтаксис**

```
match [not] destination-address-port { address-port { <ADDR>:<PORT> | <IPV6-ADDR>:<PORT> } | object-group <OBJ-GROUP-ADDRESS-PORT-NAME> | any }
```

```
no match destination-address-port
```

**Параметры**

address-port { <ADDR>:<PORT> | <IPV6-ADDR>:<PORT> } – связка IP-адресов и TCP/UDP-портов. IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255], номер порта, принимает значение [1..65535]. <IPV6-ADDR>:<PORT> – IPv6-адрес, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF], номер порта, принимает значение [1..65535];

object-group <OBJ-GROUP-ADDRESS-PORT-NAME> – имя профиля связок IP-адресов и TCP/UDP-портов, задаётся строкой до 31 символа.

При указании значения «any» правило не будет учитывать данный способ фильтрации.

**Значение по умолчанию**

any

**Необходимый уровень привилегий**

10

## Командный режим

CONFIG-SECURITY-ZONE-PAIR-RULE

### Пример

```
esr(config-security-zone-pair-rule)# match destination-address object-group local
```

### match destination-mac

Данной командой устанавливается MAC-адрес получателя, для которого должно срабатывать правило.

При использовании параметра «not» (match not) правило будет срабатывать для MAC-адресов получателя, отличных от указанного.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
match [not] destination-mac { mac <ADDR> | object-group <OBJ-GROUP-MAC-NAME> }
no match destination-mac
```

### Параметры

<ADDR> – MAC-адрес получателя, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF].

object-group <OBJ-GROUP-MAC-NAME> – имя профиля MAC-адресов, задаётся строкой до 31 символа.

При указании значения «any» правило будет срабатывать для любого MAC-адреса получателя.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-SECURITY-ZONE-PAIR-RULE

### Пример

```
esr(config-security-zone-pair-rule)# match destination-mac mac A8:F9:4B:AA:00:40
```

### match destination-nat

Данной командой устанавливается ограничение, при котором правило будет срабатывать только для трафика, измененного сервисом трансляции IP-адресов и портов получателя.

При использовании параметра «not» правило будет срабатывать для трафика, не измененного сервисом трансляции IP-адресов и портов получателя. Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
match [not] destination-nat
no match destination-nat
```

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-SECURITY-ZONE-PAIR-RULE

## Пример

```
esr(config-security-zone-pair-rule)# match destination-nat
```

## match destination-port

Данной командой устанавливается профиль TCP/UDP-портов получателя, для которых должно срабатывать правило.

При использовании параметра «not» правило будет срабатывать для TCP/UDP-портов получателя, которые не входят в указанный профиль.

Использование отрицательной формы команды (no) удаляет назначение.

## Синтаксис

```
match [not] destination-port { port-range <PORT>[-<PORT>] | object-group <PORT-SET-NAME>
| any }
no match destination-port
```

## Параметры

port-range <PORT>[-<PORT>] – address-port <PORT>[-<PORT>] – диапазон TCP/UDP-портов для правил firewall. Если не указывать TCP/UDP-порт конца диапазона, то в качестве TCP/UDP-порта для срабатывания правила используется только порт начала диапазона.

<PORT-SET-NAME> – имя профиля TCP/UDP-портов, задаётся строка до 31 символа.

При указании значения «any» правило будет срабатывать для любого TCP/UDP-порта получателя.

## Значение по умолчанию

any

## Необходимый уровень привилегий

10

## Командный режим

### CONFIG-SECURITY-ZONE-PAIR-RULE

#### Пример

```
esr(config-security-zone-pair-rule)# match destination-port object-group ssh
```

#### match fragment

Данной командой определяются фрагментированные пакеты, направленные на устройство. Команда применима только в правилах между зонами any self. Под действие правила попадают второй и последующие фрагменты пакета. Обработка пакетов этим правилом происходит до трансляции адресов DNAT.

При использовании параметра «not» правило будет срабатывать для нефраgmentированных пакетов. Использование отрицательной формы команды (no) отменяет назначение.

#### Синтаксис

```
match [not] fragment  
no match fragmen
```

#### Параметры

Отсутствуют.

#### Значение по умолчанию

Отключено.

#### Необходимый уровень привилегий

10

## Командный режим

### CONFIG-SECURITY-ZONE-PAIR-RULE

#### Пример

```
esr(config-security-zone-pair-rule)# match fragment
```

#### match icmp

Данная команда используется для настройки параметров протокола ICMP, если он выбран командой «match protocol». Данной командой устанавливается тип и код сообщений протокола ICMP, для которых должно срабатывать правило.

При использовании параметра «not» правило будет срабатывать для всех типов и кодов сообщений протокола ICMP, кроме указанных.

Использование отрицательной формы команды (no) отменяет назначение.



## Синтаксис

```
match [not] icmp { <ICMP_TYPE> <ICMP_CODE> | <OPTION> }
no match icmp
```

## Параметры

<ICMP\_TYPE> – тип сообщения протокола ICMP, принимает значения [0..255];

<ICMP\_CODE> – код сообщения протокола ICMP, принимает значения [0..255]. При указании значения «any» правило будет срабатывать для любого кода сообщения протокола ICMP;

<OPTION> – стандартные типы ICMP-сообщений, могут принимать значения:

- administratively-prohibited;
- alternate-address;
- conversion-error;
- dod-host-prohibited;
- dod-network-prohibited;
- echo;
- echo-reply;
- host-isolated;
- host-precedence;
- host-redirect;
- host-tos-redirect;
- host-tos-unreachable;
- host-unknown;
- host-unreachable;
- information-reply;
- information-request;
- mask-reply;
- mask-request;
- network-redirect;
- network-tos-redirect;
- network-tos-unreachable;
- network-unknown;
- network-unreachable;
- option-missing;
- packet-too-big;
- parameter-problem;
- port-unreachable;
- precedence;
- protocol-unreachable;
- reassembly-timeout;
- router-advertisement;
- router-solicitation;
- source-quench;
- source-route-failed;
- time-exceeded;
- timestamp-reply;
- timestamp-request;
- traceroute.

## Необходимый уровень привилегий

10

## Командный режим

### CONFIG-SECURITY-ZONE-PAIR-RULE

#### Пример

```
esr(config-security-zone-pair-rule)# match icmp 2 any
```

#### match ip-option

Данной командой определяются пакеты, содержащие опции в IP-заголовках. Команда применима только в правилах между зонами any self.

При использовании параметра «not» правило будет срабатывать для пакетов, не содержащих опций в IP-заголовках.

Использование отрицательной формы команды (no) отменяет назначение.

#### Синтаксис

```
match [not] ip-option  
no match ip-option
```

#### Параметры

Команда не содержит параметров.

#### Значение по умолчанию

Отключено.

#### Необходимый уровень привилегий

10

## Командный режим

### CONFIG-SECURITY-ZONE-PAIR-RULE

#### Пример

```
esr(config-security-zone-pair-rule)# match ip-options
```

#### match protocol

Данной командой устанавливается имя или номер IP-протокола, для которого должно срабатывать правило.

При использовании параметра «not» правило будет срабатывать для всех протоколов, кроме указанного.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
match [not] protocol <TYPE>
no match protocol
match [not] protocol-id <ID>
no match protocol-id
```

## Параметры

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre.

При указании значения «any» правило будет срабатывать для любых протоколов;

<ID> – идентификационный номер IP-протокола, принимает значения [0x00-0xFF].

## Значение по умолчанию

any

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-SECURITY-ZONE-PAIR-RULE

## Пример

```
esr(config-security-zone-pair-rule)# match protocol udp
```

## match source-address

Данной командой устанавливается профиль IP-адресов отправителя, для которых должно срабатывать правило.

При использовании параметра «not» (match not) правило будет срабатывать для IP-адресов отправителя, которые не входят в указанный профиль.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
match [not] source-address { address-range { <ADDR>[-<ADDR>] | <IPV6-ADDR>[-<IPV6-ADDR>] } | prefix { <ADDR/LEN> | <IPV6-ADDR/LEN> } | object-group <OBJ-GROUP-NETWORK-NAME> | any }
no match source-address
```

## Параметры

address-range <ADDR>[-<ADDR>] – диапазон IP-адресов для правил firewall. Если не указывать IP-адрес конца диапазона, то в качестве IP-адреса для срабатывания правила используется только IP-адрес начала диапазона.

Параметр задаётся в виде A.B.C.D, где каждая часть принимает значения [0..255]; <IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

prefix <ADDR/LEN> – IP-подсеть, используемая для срабатывания правила фильтрации firewall.

Параметр задаётся в виде A.B.C.D/E, где каждая часть A – D принимает значения [0..255] и E принимает значения [1..32]; <IPV6-ADDR/LEN> – IPv6-адрес, задаётся в виде X:X:X:X::X/E, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и E принимает значения [1..128];

object-group <OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа.

При указании значения «any» правило будет срабатывать для любого IP-адреса получателя.

### Значение по умолчанию

any

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-SECURITY-ZONE-PAIR-RULE

### Пример

```
esr(config-security-zone-pair-rule)# match source-address object-group remote
```

### match source-address-port

Данной командой устанавливается профиль связок IP-адресов и TCP/UDP-портов отправителя, для которых должно срабатывать правило.

При использовании параметра «not» (match not) правило будет срабатывать для связок IP-адресов и TCP/UDP-портов отправителя, которые не входят в указанный профиль.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
match [not] source-address-port { address-port { <ADDR>:<PORT> | <IPV6-ADDR>:<PORT> } | object-group <OBJ-GROUP-ADDRESS-PORT-NAME> | any }
```

```
no match source-address-port
```

### Параметры

address-port { <ADDR>:<PORT> | <IPV6-ADDR>:<PORT> } – связка IP-адресов и TCP/UDP-портов; IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]; номер порта, принимает значение [1..65535]; <IPV6-ADDR>:<PORT> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF]; номер порта, принимает значение [1..65535];

object-group <OBJ-GROUP-ADDRESS-PORT-NAME> – имя профиля связок IP-адресов и TCP/UDP-портов, задаётся строкой до 31 символа.

При указании значения «any» правило не будет учитывать данный способ фильтрации.

**Значение по умолчанию**

any

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-SECURITY-ZONE-PAIR-RULE

**Пример**

```
esr(config-security-zone-pair-rule)# match source-address-port object-group admin
```

**match source-mac**

Данной командой устанавливается MAC-адрес отправителя, для которого должно срабатывать правило.

При использовании параметра «not» (match not) правило будет срабатывать для MAC-адресов отправителя, отличных от указанного.

Использование отрицательной формы команды (no) отменяет назначение.

**Синтаксис**

```
match [not] source-mac { mac <ADDR> | object-group <OBJ-GROUP-MAC-NAME> }
no match source-mac
```

**Параметры**

mac <ADDR> – MAC-адрес отправителя, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF].

object-group <OBJ-GROUP-MAC-NAME> – имя профиля MAC-адресов, задаётся строкой до 31 символа.

При указании значения «any» правило будет срабатывать для любого MAC-адреса источника.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-SECURITY-ZONE-PAIR-RULE

**Пример**

```
esr(config-security-zone-pair-rule)# match source-mac mac A8:F9:4B:AA:00:40
```

## match source-port

Данной командой устанавливается профиль TCP/UDP-портов отправителя, для которых должно срабатывать правило.

При использовании параметра «not» правило будет срабатывать для TCP/UDP-портов отправителя, которые не входят в указанный профиль.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
match [not] source-port { port-range <PORT>[-<PORT>] | object-group <PORT-SET-NAME> |
any }
no match source-port
```

### Параметры

port-range <PORT>[-<PORT>] – address-port <PORT>[-<PORT>] – диапазон TCP/UDP-портов для правил firewall. Если не указывать TCP/UDP-порт конца диапазона, то в качестве TCP/UDP-порта для срабатывания правила используется только порт начала диапазона.

object-group <PORT-SET-NAME> – имя профиля TCP/UDP-портов, задаётся строка до 31 символа.

При указании значения «any» правило будет срабатывать для любого TCP/UDP-порта получателя.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-SECURITY-ZONE-PAIR-RULE

### Пример

```
esr(config-security-zone-pair-rule)# match source-port object-group telnet
```

## ports firewall enable

Данная команда активирует фильтрацию и режим отслеживания сессий при прохождении пакетов между членами Bridge-интерфейса.

Использование отрицательной формы команды (no) удаляет назначенное действие.

### Синтаксис

```
[no] ports firewall enable
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-BRIDGE

**Пример**

```
esr(config-bridge)# ports firewall enable
```

**rearrange**

Данная команда меняет шаг между созданными правилами.

**Синтаксис**

```
rearrange <VALUE>
```

**Параметры**

<VALUE> – шаг между правилами, принимает значения [1..50].

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-SECURITY-ZONE-PAIR

**Пример**

```
esr(config-security-zone-pair)# rearrange 10
```

**renumber rule**

Данная команда меняет номер правила.

**Синтаксис**

```
renumber rule <CUR_ORDER> <NEW_ORDER>
```

**Параметры**

<CUR\_ORDER> – текущий номер правила, принимает значения [1..10000];

<NEW\_ORDER> – новый номер правила, принимает значения [1..10000].

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-SECURITY-ZONE-PAIR

**Пример**

```
esr(config-security-zone-pair)# renumber rule 13 100
```

**rule**

Данная команда используется для создания правила и перехода в командный режим SECURITY ZONE PAIR RULE. Правила обрабатываются устройством в порядке возрастания их номеров.

Использование отрицательной формы команды (no) удаляет указанное правило.

**Синтаксис**

```
[no] rule <ORDER>
```

**Параметры**

<ORDER> – номер правила, принимает значения [1..10000]. Если при удалении используется значение параметра "all", то будут удалены все правила для конфигурируемой пары зон безопасности.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-SECURITY-ZONE-PAIR

**Пример**

```
esr(config-security-zone-pair)# rule 10
esr(config-security-zone-pair-rule)#
```

**security zone**

Данная команда используется для создания зон безопасности и перехода в режим конфигурирования зоны.

Использование отрицательной формы команды (no) удаляет заданную зону безопасности.

**Синтаксис**

```
[no] security zone [ <NAME> | all ]
```



**Параметры**

<NAME> – имя создаваемой зоны безопасности, задаётся строкой до 12 символов. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все зоны безопасности.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# security zone trusted
esr(config-security-zone)#
```

**security-zone**

Данная команда используется для добавления выбранного сетевого интерфейса в зону безопасности. Использование отрицательной формы команды (no) удаляет интерфейс из зоны.

**Синтаксис**

```
security-zone <NAME>
```

```
no security-zone
```

**Параметры**

<NAME> – имя зоны безопасности, задаётся строкой до 12 символов.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE  
 CONFIG-CELLULAR-MODEM  
 CONFIG-IF-E1  
 CONFIG-IF-MULTILINK  
 CONFIG-VTI  
 CONFIG-GRE  
 CONFIG-IP4IP4  
 CONFIG-LT  
 CONFIG-PPPOE  
 CONFIG-PPTP  
 CONFIG-L2TP  
 CONFIG-OPENVPN  
 CONFIG-L2TP-SERVER  
 CONFIG-OPENVPN-SERVER  
 CONFIG-PPTP-SERVER  
 CONFIG-WIREGUARD-SERVER  
 CONFIG-WIREGUARD-TUNNEL-PEER

### Пример

```
esr(config-if-gi)# security-zone trusted
```

### security zone-pair

Данная команда используется для создания группы правил для пары зон безопасности. Использование отрицательной формы команды (no) удаляет указанную группу правил.

### Синтаксис

```
[no] security zone-pair <SOURCE-ZONE> <DESTINATION-ZONE> [ vrf <VRF> ]
```

### Параметры

<SOURCE-ZONE> – зона безопасности, из которой поступает трафик. Допустимые варианты зон:

- Пользовательская - созданная пользователем;
- any - служебная зона, характеризующая любые зоны;

<DESTINATION-ZONE> – зона безопасности, в которую поступает трафик, созданная пользователем. Допустимые варианты зон:

- Пользовательская - созданная пользователем;
- any - служебная зона, описывающая любые зоны;
- self - служебная зона, описывающая трафик, предназначенный самому маршрутизатору (трафик не является транзитным);

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. Данный параметр возможно использовать только для пар зон "any"- "any" и "any"- "self";

Если при удалении используется значение параметра «all», то будут удалены все конфигурируемые пары зон безопасности.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# security zone-pair trusted self
esr(config)# security zone-pair any self vrf VRF
```

### show ip firewall counters

Данная команда используется для просмотра статистики по пакетам, проходящим между зонами, для которых не установлена сессия.

### Синтаксис

```
show ip firewall counters [ vrf <VRF> ] [ <SOURCE-ZONE> [ <DESTINATION-ZONE>
[ <ORDER> ] ] ]
```

### Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут отображены счетчики правил в указанном VRF;

<DESTINATION-ZONE> – зона безопасности, в которую поступает трафик;

<SOURCE-ZONE> – зона безопасности, из которой поступает трафик;

<ORDER> – номер правила, принимает значения [1..10000]. При указании номера правила будет показана информация только по данному правилу.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
esr# show ip firewall counters
Zone-pair          Rule          Action          Pkts          Bytes
Description
-----
any/any            default       deny            0             0             --
```

trusted/self	1	permit	0	0	From
local to router					
trusted/trusted	1	permit	0	0	--

## show ip firewall sessions

Данная команда используется для просмотра активных IP-сессий.

### Синтаксис

```
show ip firewall sessions [ vrf <VRF> ] [ protocol <TYPE> ] [ inside-source-address
<ADDR> ] [ outside-source-address <ADDR> ] [ inside-destination-address <ADDR> ]
[ outside-destination-address <ADDR> ] [ inside-source-port <PORT> ] [ outside-source-
port <PORT> ] [ inside-destination-port <PORT> ] [ outside-destination-port <PORT> ]
[ summary ] [ configuration ] [ expected ]
```

### Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут отображены активные сессии в указанном VRF;

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<PORT> – TCP/UDP-порт, принимает значения [1..65535];

inside-source-address – команда для указания IP-адреса источника входящих пакетов;

inside-destination-address – команда для указания IP-адреса назначения входящих пакетов;

outside-source-address – команда для указания IP-адреса источника отправляемых пакетов;

outside-destination-address – команда для указания IP-адреса назначения отправляемых пакетов;

inside-source-port – ключ для указания TCP/UDP-порта отправителя в входящих пакетах;

outside-source-port – ключ для указания TCP/UDP-порта отправителя в отправляемых пакетах;

inside-destination-port – ключ для указания TCP/UDP-порта назначения в входящих пакетах;

outside-destination-port – ключ для указания TCP/UDP-порта назначения в отправляемых пакетах;

summary – выводит суммарную статистику по IP-сессиям;

configuration – выводит настройку таймаутов и объема таблиц IP-сессий;

expected – команда для отображения сессий, ожидающих обработки других сессий.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
esr# show ip firewall sessions
Codes: E - expected, U - unreplied,
```

A - assured, C - confirmed

Prot	Aging	Inside source		Inside destination	Outside source
Outside	destination	Pkts	Bytes	Status	
-----	-----	-----	-----	-----	-----
icmp	22	10.0.22.3		10.0.22.15	10.0.22.3
10.0.22.15		1	84	C	
udp	19	192.168.0.15:138		192.168.0.37:138	192.168.0.15:138
192.168.0.37:138		5	1100	UC	

### show ip firewall sessions tracking

Данной командой отображается настройка функционала отслеживания сессий уровня приложений.

#### Синтаксис

```
show ip firewall sessions tracking
```

#### Параметры

Команда не содержит параметров.

#### Необходимый уровень привилегий

1

#### Командный режим

ROOT

#### Пример

```
esr# show ip firewall sessions tracking
Tracking Status:
  FTP:      Enabled
  H.323:    Enabled
  GRE:      Enabled
  PPTP:     Enabled
  NETBIOS-NS: Enabled
  SIP:      Enabled
```

### show ipv6 firewall counters

Данная команда используется для просмотра статистики по пакетам, проходящим между зонами, для которых не установлена сессия.

#### Синтаксис

```
show ipv6 firewall counters [ vrf <VRF> ] [ <SOURCE-ZONE> [ <DESTINATION-ZONE>
[ <ORDER> ] ] ]
```

## Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут отображены счетчики правил в указанном VRF;

<DESTINATION-ZONE> – зона безопасности, в которую поступает трафик;

<SOURCE-ZONE> – зона безопасности, из которой поступает трафик;

<ORDER> – номер правила, принимает значения [1..10000]. При указании номера правила будет показана информация только по данному правилу.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```

esr# show ipv6 firewall counters trusted self
Zone-pair          Rule          Action          Pkts          Bytes
Description
-----
any/any            default       deny            0             0             --
trusted/self      1             permit         0             0             From
local to router
trusted/trusted   1             permit         0             0             --

```

## show ipv6 firewall sessions

Данная команда используется для просмотра активных IPv6-сессий.

## Синтаксис

```

show ipv6 firewall sessions [ vrf <VRF> ] [summary] [ protocol <TYPE> ] [ inside-source-
address <IPV6-ADDR>] [ outside-source-address <IPV6-ADDR> ] [ inside-destination-address
<IPV6-ADDR> ] [ outside-destination-address <IPV6-ADDR> ] [ inside-source-port <PORT> ]
[ outside-source-port <PORT> ] [ inside-destination-port <PORT> ] [ outside-destination-
port <PORT> ] [ expected ] [ summary ]

```

## Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут отображены активные сессии в указанном VRF;

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;

<IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<PORT> – TCP/UDP-порт, принимает значения [1..65535];

inside-source-address – команда для указания IPv6-адреса источника входящих пакетов;

inside-destination-address – команда для указания IPv6-адреса назначения входящих пакетов;  
 outside-source-address – команда для указания IPv6-адреса источника отправляемых пакетов;  
 outside-destination-address – команда для указания IPv6-адреса назначения отправляемых пакетов;  
 inside-source-port – ключ для указания TCP/UDP-порта отправителя в входящих пакетах;  
 outside-source-port – ключ для указания TCP/UDP-порта отправителя в отправляемых пакетах;  
 inside-destination-port – ключ для указания TCP/UDP-порта назначения в входящих пакетах;  
 outside-destination-port – ключ для указания TCP/UDP-порта назначения в отправляемых пакетах;  
 expected – команда для отображения сессий, ожидающих обработки других сессий;  
 summary – выводит суммарную статистику по IPv6-сессиям.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
esr# show ipv6 firewall sessions
esr-15# show ipv6 firewall sessions
Codes: E - expected, U - unreplied,
       A - assured, C - confirmed
```

Prot	Aging	Inside source	Bytes	Inside destination	Outside source
Outside destination		Pkts		Status	
icmp6	13	fc00::2		fc00::1	fc00::2
fc00::1		--	--	C	
tcp	112	[fc00::2]:42156		[fc00::1]:22	[fc00::2]:42156
[fc00::1]:22		--	--	AC	

### show security zone

Данная команда используется для просмотра интерфейсов, входящих в зону безопасности.

### Синтаксис

```
show security zone [<NAME>]
```

### Параметры

<NAME> – имя зоны, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

1

**Командный режим**

ROOT

**Пример**

```

esr# show security zone
Zone name      Interfaces
-----
trusted        gil/0/2-6, bridge 1, openvpn(open_test)
untrusted      gil/0/1, tel/0/1-2, bridge 2, pptp(p)

```

**show security zone-pair**

Данная команда используется для просмотра списка пар зон.

**Синтаксис**

```
show security zone-pair
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример**

```

esr# show security zone-pair
From zone      To zone      VRF      Description
-----
trusted        untrusted    --        Transit zone-pair
trusted        trusted      --        --
trusted        self         WAN-2     From WAN-2
untrusted      self         WAN-1     From WAN-1

```

**show security zone-pair configuration**

Данная команда используется для просмотра правил для пары зон безопасности.

**Синтаксис**

```
show security zone-pair configuration <SOURCE-ZONE> <DESTINATION-ZONE> [<ORDER>]
```



## Параметры

<SOURCE-ZONE> – зона безопасности, из которой поступает трафик;

<DESTINATION-ZONE> – зона безопасности, в которую поступает трафик;

<ORDER> – номер правила, принимает значения [1..10000]. При указании номера правила будет показана информация только по данному правилу.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
esr# show security zone-pair configuration trusted self
Order:                               1
Description:                           --
Matching pattern:
  Protocol:                             icmp
  Fragment:
  IP options:
  Source MAC:                             any
  Destination MAC:                         any
  Source address:                          10.0.34.12
  Destination address:                      10.0.34.90
  Destination NAT:                          --
  Application:                              --
Action:                                  Permit
Status:                                  Enabled
-----
```

## Управление логированием и защитой от сетевых атак

- Управление защитой от сетевых атак
  - `ip firewall screen dos-defense icmp-threshold`
  - `ip firewall screen dos-defense land`
  - `ip firewall screen dos-defense limit-session-destination`
  - `ip firewall screen dos-defense limit-session-source`
  - `ip firewall screen dos-defense syn-flood`
  - `ip firewall screen dos-defense udp-threshold`
  - `ip firewall screen dos-defense winnuke`
  - `ip firewall screen spy-blocking fin-no-ack`
  - `ip firewall screen spy-blocking icmp-type destination-unreachable`
  - `ip firewall screen spy-blocking icmp-type echo-request`
  - `ip firewall screen spy-blocking icmp-type reserved`
  - `ip firewall screen spy-blocking icmp-type source-quench`
  - `ip firewall screen spy-blocking icmp-type time-exceeded`
  - `ip firewall screen spy-blocking ip-sweep`
  - `ip firewall screen spy-blocking port-scan`
  - `ip firewall screen spy-blocking spoofing`
  - `ip firewall screen spy-blocking syn-fin`
  - `ip firewall screen spy-blocking tcp-all-flags`
  - `ip firewall screen spy-blocking tcp-no-flag`
  - `ip firewall screen suspicious-packets icmp-fragment`
  - `ip firewall screen suspicious-packets ip-fragment`
  - `ip firewall screen suspicious-packets large-icmp`
  - `ip firewall screen suspicious-packets syn-fragment`
  - `ip firewall screen suspicious-packets udp-fragment`
  - `ip firewall screen suspicious-packets unknown-protocols`
- Управление оповещением о сетевых атаках
  - `ip firewall logging interval`
  - `logging firewall screen detailed`
  - `logging firewall screen dos-defense`
  - `logging firewall screen spy-blocking`
  - `logging firewall screen suspicious-packets`
  - `show ip firewall screens counters`

## Управление защитой от сетевых атак

### `ip firewall screen dos-defense icmp-threshold`

Данная команда включает защиту от ICMP flood атак. При включенной защите ограничивается количество icmp-пакетов всех типов в секунду для одного адреса назначения.

Использование отрицательной формы команды (`no`) отключает защиту от ICMP flood атак.

#### Синтаксис

```
ip firewall screen dos-defense icmp-threshold <NUM>
no ip firewall screen dos-defense icmp-threshold
```

#### Параметры

<NUM> – количество icmp-пакетов в секунду задается в диапазоне [1..10000].

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip firewall screen dos-defense icmp-threshold 2000
```

**ip firewall screen dos-defense land**

Данная команда включает защиту от land-атак. При включенной защите блокируются пакеты с одинаковыми source и destination ip, и флагом SYN в заголовке TCP.

Использование отрицательной формы команды (no) отключает защиту от land-атак.

**Синтаксис**

```
[no] ip firewall screen dos-defense land
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip firewall screen dos-defense land
```

**ip firewall screen dos-defense limit-session-destination**

Когда таблица IP-сессий хоста переполняется, он больше не в состоянии организовывать новые сессии и отбрасывает запросы (такое может происходить при различных DoS-атаках: SYN flood, UDP flood, ICMP flood, и т. д.). Команда включает ограничение числа пакетов, передаваемых за секунду на один адреса назначения, которое смягчает DoS-атаки.

Использование отрицательной формы команды (no) снимает это ограничение.

### Синтаксис

```
ip firewall screen dos-defense limit-session-destination <NUM>  
no ip firewall screen dos-defense limit-session-destination
```

### Параметры

<NUM> – ограничение количества ip-пакетов в секунду, задается в диапазоне [1..10000].

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

### Пример

```
esr(config)# ip firewall screen dos-defense limit-session-destination 1000
```

### ip firewall screen dos-defense limit-session-source

Когда таблица IP-сессий хоста переполняется, он больше не в состоянии организовывать новые сессии и отбрасывает запросы (такое может происходить при различных DoS-атаках: SYN flood, UDP flood, ICMP flood, и т.д.). Команда включает ограничение числа пакетов, передаваемых за секунду с одного адреса источника, которое смягчает DoS-атаки.

Использование отрицательной формы команды (no) снимает это ограничение.

### Синтаксис

```
ip firewall screen dos-defense limit-session-source <NUM>  
no ip firewall screen dos-defense limit-session-source
```

### Параметры

<NUM> – ограничение количества ip-пакетов в секунду задается в диапазоне [1..10000].

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

## Пример

```
esr(config)# ip firewall screen dos-defense limit-session-source 1000
```

## ip firewall screen dos-defense syn-flood

Данная команда включает защиту от SYN flood атак. При включенной защите ограничивается количество TCP-пакетов с установленным флагом SYN в секунду для одного адреса назначения.

Использование отрицательной формы команды (no) отключает защиту от SYN flood атак.

### Синтаксис

```
ip firewall screen dos-defense syn-flood <NUM> [src-dst]
```

```
no ip firewall screen dos-defense syn-flood
```

### Параметры

<NUM> – максимальное количество TCP-пакетов с установленным флагом SYN в секунду задается в диапазоне [1..10000].

src-dst – ограничение количества TCP-пакетов с установленным флагом SYN на основании адреса источника и адреса назначения.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

## Пример

```
esr(config)# ip firewall screen dos-defense syn-flood 100 src-dsr
```

## ip firewall screen dos-defense udp-threshold

Данная команда включает защиту от UDP flood атак. При включенной защите ограничивается количество UDP-пакетов в секунду для одного адреса назначения.

Использование отрицательной формы команды (no) отключает защиту от UDP flood.

### Синтаксис

```
ip firewall screen dos-defense udp-threshold <NUM>
```

```
no ip firewall screen dos-defense udp-threshold
```

### Параметры

<NUM> – максимальное количество UDP-пакетов в секунду, задается в диапазоне [1..10000].

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip firewall screen dos-defense udp-threshold
```

**ip firewall screen dos-defense winnuke**

Данная команда включает защиту от winnuke-атак. При включенной защите блокируются TCP-пакеты с установленным флагом URG и 139 портом назначения.

Использование отрицательной формы команды (no) отключает защиту от winnuke-атак.

**Синтаксис**

```
[no] ip firewall screen dos-defense winnuke
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip firewall screen dos-defense winnuke
```

**ip firewall screen spy-blocking fin-no-ack**

Данная команда включает блокировку TCP-пакетов с установленным флагом FIN и не установленным флагом ACK.

Использование отрицательной формы команды (no) отключает блокировку TCP-пакетов с установленным флагом FIN и не установленным флагом ACK.

**Синтаксис**

```
[no] ip firewall screen spy-blocking fin-no-ack
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip firewall screen spy-blocking fin-no-ack
```

**ip firewall screen spy-blocking icmp-type destination-unreachable**

Данная команда включает блокировку всех ICMP-пакетов 3 типа (destination-unreachable), включая пакеты, сгенерированные самим маршрутизатором.

Использование отрицательной формы команды (no) отключает блокировку ICMP-пакетов 3 типа.

**Синтаксис**

```
[no] ip firewall screen spy-blocking icmp-type destination-unreachable
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip firewall screen spy-blocking icmp-type destination-unreachable
```

**ip firewall screen spy-blocking icmp-type echo-request**

Данная команда включает блокировку всех ICMP пакетов 8 типа (echo-request), включая пакеты, сгенерированные самим маршрутизатором.

Использование отрицательной формы команды (no) отключает блокировку ICMP-пакетов 8 типа.

**Синтаксис**

```
[no] ip firewall screen spy-blocking icmp-type echo-request
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip firewall screen spy-blocking icmp-type echo-request
```

**ip firewall screen spy-blocking icmp-type reserved**

Данная команда включает блокировку всех ICMP-пакетов 2 и 7 типов (reserved), включая пакеты, сгенерированные самим маршрутизатором.

Использование отрицательной формы команды (no) отключает блокировку ICMP-пакетов 2 и 7 типов.

**Синтаксис**

```
[no] ip firewall screen spy-blocking icmp-type reserved
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG



## Пример

```
esr(config)# ip firewall screen spy-blocking icmp-type reserved
```

### **ip firewall screen spy-blocking icmp-type source-quench**

Данная команда включает блокировку всех ICMP-пакетов 4 типа (source quench), включая пакеты, сгенерированные самим маршрутизатором.

Использование отрицательной формы команды (no) отключает блокировку ICMP-пакетов 4 типа.

#### **Синтаксис**

```
[no] ip firewall screen spy-blocking icmp-type source-quench
```

#### **Параметры**

Команда не содержит параметров.

#### **Значение по умолчанию**

Отключено.

#### **Необходимый уровень привилегий**

15

#### **Командный режим**

CONFIG

## Пример

```
esr(config)# ip firewall screen spy-blocking icmp-type source-quench
```

### **ip firewall screen spy-blocking icmp-type time-exceeded**

Данная команда включает блокировку всех ICMP-пакетов 11 типа (time exceeded), включая пакеты, сгенерированные самим маршрутизатором.

Использование отрицательной формы команды (no) отключает блокировку ICMP-пакетов 11 типа.

#### **Синтаксис**

```
[no] ip firewall screen spy-blocking icmp-type time-exceeded
```

#### **Параметры**

Команда не содержит параметров.

#### **Значение по умолчанию**

Отключено.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# ip firewall screen spy-blocking icmp-type time-exceeded
```

## ip firewall screen spy-blocking ip-sweep

Данная команда включает защиту от IP-sweep атак. При включенной защите, если в течение заданного в параметрах интервала приходит более 10 ICMP-запросов от одного источника, первые 10 запросов пропускаются маршрутизатором, а 11 и последующие отбрасываются на оставшееся время интервала.

Использование отрицательной формы команды (no) отключает защиту от ip-sweep атак.

## Синтаксис

```
ip firewall screen spy-blocking ip-sweep <THRESHOLD> [ <TIME> ]
```

```
no ip firewall screen spy-blocking ip-sweep
```

## Параметры

<THRESHOLD> – количество пакетов ip sweep атаки в секунду, задается в диапазоне [1..10000].

<TIME> – время блокировки в миллисекундах [1..1000000].

## Значение по умолчанию

Отключено.

Без указания времени блокировки при включении устанавливается значение 10000.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# ip firewall screen spy-blocking ip-sweep 1000
```

## ip firewall screen spy-blocking port-scan

Данная команда включает защиту от port scan атак. Если в течение секунды на один источник приходит более <THRESHOLD> TCP-пакетов с флагом SYN на разные TCP-порты или более <THRESHOLD> UDP-пакетов на разные UDP-порты, то такое поведение фиксируется как port scan атака, и все последующие пакеты такого рода от источника блокируются на второй заданный интервал времени (<TIME>).

Использование отрицательной формы команды (no) отключает защиту от port scan атак.

### Синтаксис

```
ip firewall screen spy-blocking port-scan <THRESHOLD> [ <TIME> ]
no ip firewall screen spy-blocking port-scan
```

### Параметры

<THRESHOLD> – количество пакетов port scan атаки в секунду, указывается в диапазоне [1..10000].

<TIME> – время блокировки атакующего узла в миллисекундах [1..1000000].

### Значение по умолчанию

Отключено.

Без указания времени блокировки при включении, устанавливается значение 10000.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

### Пример

```
esr(config)# ip firewall screen spy-blocking port-scan 100 1000
```

### ip firewall screen spy-blocking spoofing

Данная команда включает защиту от IP spoofing атак. При включенной защите маршрутизатор проверяет пакеты на соответствие адреса источника и записей в таблице маршрутизации, и в случае несоответствия пакет отбрасывается. Например, если пакет с адресом источника 10.0.0.1/24 приходит на интерфейс Gi1/0/1, а в таблице маршрутизации данная подсеть располагается за интерфейсом Gi1/0/2, то считается, что адрес источника был подменен.

Использование отрицательной формы команды (no) отключает защиту от ip spoofing атак.

### Синтаксис

```
[no] ip firewall screen spy-blocking spoofing
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

15

## Командный режим

CONFIG

### Пример

```
esr(config)# ip firewall screen spy-blocking spoofing
```

#### **ip firewall screen spy-blocking syn-fin**

Данная команда включает блокировку TCP-пакетов, с установленными флагами SYN и FIN.

Использование отрицательной формы команды (no) отключает блокировку TCP-пакетов, с установленными флагами SYN и FIN.

#### Синтаксис

```
[no] ip firewall screen spy-blocking syn-fin
```

#### Параметры

Команда не содержит параметров.

#### Значение по умолчанию

Отключено.

#### Необходимый уровень привилегий

15

## Командный режим

CONFIG

### Пример

```
esr(config)# ip firewall screen spy-blocking syn-fin
```

#### **ip firewall screen spy-blocking tcp-all-flags**

Данная команда включает блокировку TCP-пакетов, со всеми флагами или с набором флагов: FIN, PSH, URG. Обеспечивается защита от атаки XMAS.

Использование отрицательной формы команды (no) отключает блокировку TCP-пакетов, со всеми флагами или с набором флагов: FIN,PSH,URG.

#### Синтаксис

```
[no] ip firewall screen spy-blocking tcp-all-flag
```

#### Параметры

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip firewall screen spy-blocking tcp-all-flag
```

**ip firewall screen spy-blocking tcp-no-flag**

Данная команда включает блокировку TCP-пакетов, с нулевым полем flags.

Использование отрицательной формы команды (no) отключает блокировку TCP-пакетов, с нулевым полем flags.

**Синтаксис**

```
[no] ip firewall screen spy-blocking tcp-no-flag
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip firewall screen spy-blocking tcp-no-flag
```

**ip firewall screen suspicious-packets icmp-fragment**

Данная команда включает блокировку фрагментированных ICMP-пакетов.

Использование отрицательной формы команды (no) отключает блокировку фрагментированных ICMP-пакетов.

**Синтаксис**

```
[no] ip firewall screen suspicious-packets icmp-fragment
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip firewall screen suspicious-packets icmp-fragment
```

**ip firewall screen suspicious-packets ip-fragment**

Данная команда включает блокировку фрагментированных IP-пакетов.

Использование отрицательной формы команды (no) отключает блокировку фрагментированных пакетов.

**Синтаксис**

```
[no] ip firewall screen suspicious-packets ip-fragment
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip firewall screen suspicious-packets ip-fragment
```

**ip firewall screen suspicious-packets large-icmp**

Данная команда включает блокировку ICMP-пакетов длиной более 1024 байт.

Использование отрицательной формы команды (no) отключает блокировку ICMP-пакетов длиной более 1024 байт.

**Синтаксис**

```
[no] ip firewall screen suspicious-packets large-icmp
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip firewall screen suspicious-packets large-icmp
```

**ip firewall screen suspicious-packets syn-fragment**

Данная команда включает блокировку фрагментированных TCP-пакетов, с флагом SYN.

Использование отрицательной формы команды (no) отключает блокировку фрагментированных TCP-пакетов, с флагом SYN.

**Синтаксис**

```
[no] ip firewall screen suspicious-packets syn-fragment
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

## Пример

```
esr(config)# ip firewall screen suspicious-packets syn-fragment
```

## **ip firewall screen suspicious-packets udp-fragment**

Данная команда включает блокировку фрагментированных UDP-пакетов.

Использование отрицательной формы команды (no) отключает блокировку фрагментированных UDP-пакетов.

### Синтаксис

```
[no] ip firewall screen suspicious-packets udp-fragment
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

## Пример

```
esr(config)# ip firewall screen suspicious-packets udp-fragment
```

## **ip firewall screen suspicious-packets unknown-protocols**

Данная команда включает блокировку пакетов, с ID протокола в заголовке IP равном 137 и более.

Использование отрицательной формы команды (no) отключает блокировку пакетов, с ID протокола в заголовке IP равном 137 и более.

### Синтаксис

```
[no] ip firewall screen suspicious-packets unknown-protocols
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено.



**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip firewall screen suspicious-packets unknown-protocols
```

**Управление оповещением о сетевых атаках****ip firewall logging interval**

Данной командой устанавливается частота оповещения (по SNMP, syslog и в CLI) об обнаруженных и отраженных сетевых атаках. При детектировании атаки сообщение логируется мгновенно, но следующие оповещения о данной конкретной атаке будут логироваться через заданный интервал времени, если атака будет носить продолжительный характер.

Использование отрицательной формы команды (no) возвращает таймер к значению по умолчанию.

**Синтаксис**

```
ip firewall logging interval <NUM>
```

```
no ip firewall logging interval
```

**Параметры**

<NUM> – интервал времени в секундах [30 .. 2147483647].

**Значение по умолчанию**

30

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip firewall logging interval 60
```

**logging firewall screen detailed**

Данной командой включается более детальный вывод сообщений по обнаруженным и отраженным сетевым атакам в CLI.

В детальном выводе, кроме имени интерфейса, с которого пришёл пакет, отображается IP-адрес источника и IP-адрес назначения этого пакета, а также MAC-адрес источника, отправившего этот пакет. Использование отрицательной формы команды (no) отключает детальный вывод сообщений.

### Синтаксис

```
[no] logging firewall screen detailed
```

### Параметры

Отсутствуют.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

### Пример

```
esr(config)# logging firewall screen detailed
```

### logging firewall screen dos-defense

Данной командой включается механизм обнаружения и логирования DoS-атак через CLI, syslog и по SNMP. В связке с включенной защитой от атак будет производиться оповещение об отраженных DoS-атаках.

Использование отрицательной формы команды (no) отключает механизм обнаружения и логирования обнаруженных и отраженных DoS-атак.

### Синтаксис

```
[no] logging firewall screen dos-defense <ATAK_TYPE>
```

### Параметры

<ATAK\_TYPE> – тип DoS-атаки, принимает значения:

- icmp-threshold;
- land;
- limit-session-destination;
- limit-session-source;
- syn-flood;
- udp-threshold;
- winnuke.

Подробное описание DoS-атак приведено в разделе [Управление защитой от сетевых атак](#).

### Значение по умолчанию

Отключено.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# logging firewall screen dos-defense land
```

## logging firewall screen spy-blocking

Данной командой включается механизм обнаружения и логирования шпионской активности через CLI, syslog и по SNMP. В связке с включенной защитой от шпионской активности будет производиться оповещение о блокированной шпионской активности.

Использование отрицательной формы команды (no) отключает механизм обнаружения и логирования обнаруженной и блокированной шпионской активности.

## Синтаксис

```
[no] logging firewall screen spy-blocking { <ATAK_TYPE> | icmp-type <ICMP_TYPE> }
```

## Параметры

<ATAK\_TYPE> – тип шпионской активности, принимает значения:

- fin-no-ack;
- ip-sweep;
- port-scan;
- spoofing;
- syn-fin;
- tcp-all-flag;
- tcp-no-flag.

<ICMP\_TYPE> – тип icmp, принимает значения:

- destination-unreachable;
- echo-request;
- reserved;
- source-quench;
- time-exceeded.

Подробное описание шпионских активностей приведено в разделе [Управление защитой от сетевых атак](#).

## Значение по умолчанию

Отключено.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# logging firewall screen spy-blocking icmp-type echo-request
```

## logging firewall screen suspicious-packets

Данной командой включается механизм обнаружения нестандартных пакетов и логирования через CLI, syslog и по SNMP. В связке с включенной защитой от нестандартных пакетов будет производиться также оповещение о блокировке нестандартных пакетов.

Использование отрицательной формы команды (no) отключает механизм обнаружения и логирования обнаруженных и заблокированных нестандартных пакетов.

## Синтаксис

```
[no] logging firewall screen suspicious-packets <PACKET_TYPE>
```

## Параметры

< PACKET\_TYPE> – тип нестандартных пакетов, принимает значения:

- icmp-fragment;
- ip-fragment;
- large-icmp;
- syn-fragment;
- udp-fragment;
- unknown-protocols.

Подробное описание защиты от нестандартных пакетов приведено в разделе [Управление защитой от сетевых атак](#).

## Значение по умолчанию

Отключено.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# logging firewall screen suspicious-packets icmp-fragment
```

## show ip firewall screens counters

Данная команда позволяет просматривать статистику по обнаруженным сетевым атакам.

## Синтаксис

```
show ip firewall screens counters
```

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

10

## Командный режим

ROOT

## Пример

```
esr# show ip firewall screens counters
DDoS:
  Destination limit screen:  --
  Source limit screen:      --
  ICMP threshold screen:    1
  UDP threshold screen:     --
  SYN flood screen:         0
  Land attack screen:       1
  Winnuke attack screen:    --
Suspicious packets:
  ICMP fragmented screen:   --
  UDP fragmented screen:    --
  Large ICMP screen:        4
  Fragmented SYN screen:    --
  Unknown protocol screen:  --
  Fragmented IP screen:     --
Spying:
  Port scanning screen:     --
  IP sweep screen:          --
  SYN-FIN screen:           --
  TCP all flags screen:     --
  FIN no ACK screen:        --
  TCP no flags screen:      --
  Spoofing screen:          --
ICMP types:
  ICMP reserved screen:     --
  ICMP quench screen:       --
  ICMP echo request screen: --
  ICMP time exceeded screen: --
  ICMP unreachable screen:  --
```

## Управление фильтрацией

- [default action](#)
- [description](#)
- [filter](#)
- [log enable](#)
- [ip http profile](#)
- [ip http proxy](#)
- [ip http proxy aaa das-profile](#)
- [ip http proxy listen-ports](#)
- [ip http proxy redirect-port](#)
- [ip http proxy server-url](#)
- [ip https proxy](#)
- [ip https proxy listen-ports](#)
- [ip https proxy redirect-port](#)
- [urls local](#)
- [urls remote](#)

### default action

Данной командой устанавливается действие для обработки трафика неавторизованных пользователей.

Использование отрицательной формы команды (no) удаляет действие для обработки трафика неавторизованных пользователей.

### Синтаксис

```
default action <ACTION> { redirect-url <REDIRECT-URL>}
```

```
no default action
```

### Параметры

<ACTION> – действие, которое будет происходить для трафика, никак не описанного в профиле.

Возможные значения:

- permit – пропускать http-запросы;
- deny – блокировать http-запросы;
- redirect – перенаправлять запросы на определенный URL-адрес.

<REDIRECT-URL> – URL-адрес для перенаправления http-запросов. Указывается только в режиме redirect.

### Значение по умолчанию

Не задано.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-PROXY-PROFILE

## Пример

```
esr(config-profile)# default action deny
```

## description

Данной командой задаётся описание профиля фильтрации.

Использование отрицательной формы команды (no) удаляет описание.

## Синтаксис

```
description <DESCRIPTION>
```

```
no description
```

## Параметры

<DESCRIPTION> – описание, задаётся строкой до 255 символов.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-PROXY-PROFILE

## Пример

```
esr(config-profile)# description "deny any"
```

## filter

Данной командой включается режим фильтрации данных.

Использование отрицательной формы команды (no) режим фильтрации данных отключается.

## Синтаксис

```
[no] filter <DATA-TYPE>
```

## Параметры

<DATA-TYPE> – тип данных, подлежащих фильтрации. Может принимать значения:

- activex;
- cookie;
- js.

## Значение по умолчанию

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-PROXY-PROFILE

**Пример**

```
esr(config-profile)# filter cookie
```

**log enable**

Данной командой включается ведение журнала профиля прокси-сервера.

Использование отрицательной формы команды (no) отключает ведение журнала.

**Синтаксис**

```
[no] log enable
```

**Параметры**

Отсутствуют.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-PROXY-PROFILE

**Пример**

```
esr(config-proxy-profile)# log enable
```

**ip http profile**

Данной командой создается профиль фильтрации HTTP-трафика и производится переход в режим конфигурирования созданного профиля.

Использование отрицательной формы команды (no) удаляет профиль фильтрации HTTP-трафика.

**Синтаксис**

```
[no] ip http profile <NAME>
```



## Параметры

<NAME> – имя профиля фильтрации HTTP-трафика, задается строкой длиной [1..31] символ.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# ip http profile HTTP_BLOCKED
```

## ip http proxy

Данной командой на интерфейс или туннель назначается профиль фильтрации http-запросов. Использование отрицательной формы команды (no) удаляет профиль фильтрации http-запросов.

## Синтаксис

```
ip http proxy <NAME>
```

```
no ip http proxy
```

## Параметры

<NAME> – имя профиля, задается строкой до 31 символа.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-CELLULAR-MODEM

CONFIG-GRE

CONFIG-IP4IP4  
CONFIG-VTI  
CONFIG-LT  
CONFIG-PPPOE

### Пример

```
esr(config-cellular-modem)# ip http proxy HTTP-BLOCKED
```

### **ip http proxy aaa das-profile**

Данная команда используется для выбора профиля серверов динамической авторизации (DAS), на которые будут приходить CoA-запросы от PCRF об изменении списков URL.

Использование отрицательной формы команды (no) удаляет профиль динамической авторизации.

### Синтаксис

```
ip http proxy aaa das-profile <NAME>  
no ip http proxy aaa das-profile
```

### Параметры

<NAME> – имя профиля серверов динамической авторизации (DAS), задается строкой до 31 символа.

### Значение по умолчанию

Профиль не задан.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

### Пример

```
esr(config)# ip http proxy aaa das-profile my_server
```

### **ip http proxy listen-ports**

Данной командой устанавливается список tcp-портов для фильтрации http-запросов (вместо стандартных 80 и 8080).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
[no] ip http proxy listen-ports <SERVICE_OBJ_GROUP_NAME>
```

## Параметры

<SERVICE\_OBJ\_GROUP\_NAME> – профиль портов (object-group service). Задается строкой до 31 символа.

## Значение по умолчанию

80, 8080

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# ip http proxy listen-ports HTTP-ADD
```

## ip http proxy redirect-port

Данной командой определяется базовый порт HTTP Proxy-сервера на маршрутизаторе. Количество используемых портов соответствует числу ядер процессора.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
ip http proxy redirect-port <PORT>
```

```
no ip http proxy redirect-port
```

## Параметры

<PORT> – номер порта, указывается в диапазоне [1..65535].

## Значение по умолчанию

3128

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# ip proxy http redirect-port 3001
```

**ip http proxy server-url**

Данной командой задается URL-адрес сервера, с которого будет загружаться список фильтрации. Использование отрицательной формы команды (no) удаляет URL-адрес сервера.

**Синтаксис**

```
[no] ip http proxy server-url <URL>
```

**Параметры**

<URL> – URL-адрес ресурса, откуда необходимо скачивать файл со списком фильтрации. Задается строкой длиной [8..255] символов.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip http proxy server-url http://domain.example
```

**ip https proxy**

Данной командой на интерфейс или туннель назначается профиль фильтрации https-запросов. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
ip https proxy <NAME>
```

```
no ip https proxy
```

**Параметры**

<NAME> – имя профиля. Задается строкой до 31 символа.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-CELLULAR-MODEM  
CONFIG-GRE  
CONFIG-IP4IP4  
CONFIG-VTI  
CONFIG-LT  
CONFIG-PPPOE

### Пример

```
esr(config-cellular-modem)# ip https proxy HTTPS-BLOCKED
```

### ip https proxy listen-ports

Данной командой устанавливается список дополнительных tcp-портов для фильтрации https-запросов (в дополнение к стандартному 433).

Использование отрицательной формы команды (no) удаляет ранее созданный список.

### Синтаксис

```
[no] ip https proxy listen-ports <SERVICE_OBJ_GROUP_NAME>
```

### Параметры

<SERVICE\_OBJ\_GROUP\_NAME> – профиль портов (object-group service). Задается строкой до 31 символа.

### Значение по умолчанию

433

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# ip https proxy listen-ports HTTPS-ADD
```

**ip https proxy redirect-port**

Данной командой определяется базовый порт HTTPS Proxy-сервера на маршрутизаторе. Количество используемых портов соответствует числу ядер процессора.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
ip https proxy redirect-port <PORT>
no ip https proxy redirect-port
```

**Параметры**

<PORT> – номер порта, указывается в диапазоне [1..65535].

**Значение по умолчанию**

3129

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip proxy https redirect-port 3001
```

**urls local**

Данной командой в профиль фильтрации добавляется локальный список URL-адресов и действий для данного списка.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
urls local <URL_OBJ_GROUP_NAME> action { deny | permit | redirect redirect-url <REDIRECT-URL> }
no urls local
```

**Параметры**

<URL\_OBJ\_GROUP\_NAME> – список URL, для которых будет применяться действие;

permit – пропускать http-запросы на адреса, описанные в указанном списке URL;

deny – блокировать http-запросы на адреса, описанные в указанном списке URL;

redirect – перенаправлять запросы на адреса, описанные в указанном списке URL на определенный URL-адрес;

<REDIRECT-URL> – URL-адрес для перенаправления http-запросов. Указывается только в режиме redirect.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-PROFILE

### Пример

```
esr(config-profile)# urls local OGU_DENY action deny
```

### urls remote

Данной командой в профиль фильтрации добавляется ссылка на удаленный список URL-адресов и действий для данного списка.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
urls remote <FILE-NAME> action { deny | permit | redirect redirect-url <REDIRECT-URL> }  
no urls remote
```

### Параметры

<FILE-NAME> – имя файла, содержащее список URL и расположенного на сервере, прописанном в команде ip http proxy server-url (см. раздел [ip http proxy server-url](#)). Имя файла задается строкой до 31 символов;

permit – пропускать http-запросы на адреса, описанные в файле с указанным именем;

deny – блокировать http-запросы на адреса, описанные в файле с указанным именем;

redirect – перенаправлять запросы на адреса, описанные в файле с указанным именем на определенный URL-адрес.

<REDIRECT-URL> – URL-адрес для перенаправления http-запросов. Указывается только в режиме redirect.

### Необходимый уровень привилегий

15


### Командный режим

CONFIG-PROFILE

### Пример

```
esr(config-profile)# urls remote http-deny action deny
```

## Управление системой предотвращения вторжений (IPS/IDS)

 Данный функционал активируется только при наличии лицензии.

- Общие команды IPS/IDS
  - clear content-filter cache
  - clear security ips counters
  - description
  - enable
  - show content-filter cache
  - show security ips content-provider
  - show security ips content-provider rules-info
  - show security ips counters
  - show security ips counters application-layer
  - show security ips counters decoder
  - show security ips counters flow
  - show security ips counters general
  - show security ips counters tcp
  - show security ips status
  - show security ips user-server
  - update security ips content-provider rules
  - update security ips content-provider rules-info
  - update security ips user-server rules
- Настройка политики IPS/IDS
  - category
  - external network-group
  - protect network-group
  - rules
  - rules action
  - security ips policy
  - vendor
- Настройка IPS
  - fail-close enable
  - logging ips severity
  - logging remote-server
  - logging update-interval
  - performance max
  - policy
  - queue-limit
  - default-packet-size follow-mtu
  - security ips
  - service-ips
- Настройка параметров кэширования записей контентной фильтрации
  - content-filter
  - uri cache-size
  - uri reachable-interval
- Настройка автообновления правил IPS/IDS, распространяемых по коммерческой лицензии
  - content-provider
  - host address
  - host port
  - location
  - reboot
  - storage-path
  - system-name
  - upgrade interval



- Настройка автообновления правил IPS/IDS из внешних источников
  - auto-upgrade
  - storage-path
  - upgrade interval
  - url
  - user-server
- Настройка пользовательских правил IPS/IDS
  - action
  - destination-address
  - destination-port
  - direction
  - ip dscp
  - ip ftp command
  - ip ftp-data command
  - ip http
  - ip http content-filter
  - ip http method
  - ip icmp code
  - ip icmp code comparison-operator
  - ip icmp id
  - ip icmp sequence-id
  - ip icmp type
  - ip icmp type comparison-operator
  - ip protocol-id
  - ip tcp acknowledgment-number
  - ip tcp sequence-id
  - ip tcp window-size
  - ip tls content-filter
  - ip ttl
  - ip ttl comparison-operator
  - meta classification-type
  - meta log-message
  - payload content
  - payload data-size
  - payload data-size comparison-operator
  - payload depth
  - payload no-case
  - payload offset
  - protocol
  - rule
  - security ips-category user-defined
  - source-address
  - source-port
  - threshold count
  - threshold second
  - threshold track
  - threshold type
- Настройка расширенных пользовательских правил
  - rule-advanced
  - rule-text

## Общие команды IPS/IDS

### clear content-filter cache

Данной командой выполняется очистка кэшированных ответов от Kaspersky-lab.

#### Синтаксис

```
clear content-filter cache [ url <URL> ]
```

#### Параметры

<URL> – URL, который необходимо удалить из кэша, задаётся строкой от 4 до 255 символов.

#### Необходимый уровень привилегий

10

#### Командный режим

ROOT

#### Пример

```
esr# clear content-filter cache url eltex-co.ru
```

### clear security ips counters

Данной командой выполняется сброс счетчиков работы правил сервиса IPS/IDS.

#### Синтаксис

```
clear security ips counters
```

#### Необходимый уровень привилегий

15

#### Командный режим

ROOT

#### Пример

```
esr# clear security ips counters
esr# 15: 2023-09-11T12:21:00.000+07:00 %IPS-I-INFO: Counters have been cleared completely !
```

### description

Данной командой выполняется изменение описания.

Использование отрицательной формы команды (no) удаляет описание.

**Синтаксис**

description <DESCRIPTION>

no description

**Параметры**

<DESCRIPTION> – описание, задаётся строкой до 255 символов.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IPS-CATEGORY

CONFIG-IPS-CATEGORY-RULE

CONFIG-IPS-CATEGORY-RULE-ADVANCED

CONFIG-IPS-POLICY

CONFIG-IPS-UPGRADE-USER-SERVER

CONFIG-CONTENT-PROVIDER

**Пример**

```
esr(config-ips-upgrade-user-server)# description "Etnetera aggressive IP blacklist"
```

**enable**

Данной командой активируется сервис IPS/IDS и его правила.

Использование отрицательной формы команды (no) деактивирует сервис IPS/IDS.

**Синтаксис**

[no] enable

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

IPS/IDS-сервис не активирован.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IPS

CONFIG-IPS-CATEGORY-RULE

CONFIG-IPS-CATEGORY-RULE-ADVANCED

CONFIG-CONTENT-PROVIDER  
 CONFIG-IPS-UPGRADE-USER-SERVER

### Пример

```
esr(config-ips)# enable
```

### show content-filter cache

Данной командой выполняется просмотр записей контентной фильтрации, расположенных в кэше.

### Синтаксис

```
show content-filter cache [url <URL> | vendor <NAME>]
```

### Параметры

url <URL> – просмотр записей, отфильтрованных по URL. <URL> – текстовое поле, содержащее URL-ссылку длиной от 4 до 255 символов.

vendor <NAME> – просмотр записей, отфильтрованных по поставщику категорий. <NAME> – название поставщика категорий контентной фильтрации. Принимает значения:

Kaspersky-Lab – в текущей версии ПО в качестве поставщика категорий контентной фильтрации может выступать только Лаборатория Касперского.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример

```
esr# show content-filter cache
URI                               Vendor                Categories                Time to live
-----
rutube.ru                         kaspersky-lab        downloadable-content,    06,23:28:32
                               hobbies-recreation,
                               media-content
ya.ru                              kaspersky-lab        it-services, searchers   06,23:29:04
vtb.ru                             kaspersky-lab        finance, human-life,     06,23:46:01
                               online-banks, payments,
                               transactions
sber.ru                            kaspersky-lab        online-banks, payments   06,23:54:49
youtube.com                       kaspersky-lab        downloadable-content,    06,23:54:57
                               media-content
```

### show security ips content-provider

Данной командой выполняется просмотр информации об обновлении правил IPS/IDS, распространяемых по коммерческой лицензии.

**Синтаксис**

```
show security ips content-provider
```

**Необходимый уровень привилегий**

10

**Командный режим**

ROOT

**Пример**

```
esr# show security ips content-provider
Server: content-provider
Last MD5 of received files:          93633ab9a73248ea50d58c25b1ac806c
Next update: 2020-10-06 12:27:40
```

**show security ips content-provider rules-info**

Данной командой выполняется просмотр информации о категориях правил IPS/IDS, доступных по текущей коммерческой лицензии. При отсутствии действующей лицензии список будет пуст.

**Синтаксис**

```
show security ips content-provider rules-info
```

**Необходимый уровень привилегий**

10

**Командный режим**

ROOT

**Пример**

```
esr# show security ips content-provider rules-info
Vendor : kaspersky
  Category : IoURLsDF
    Count of rules : 8000
    Description : Kaspersky Lab IoURLsDF feed
                  IoURLsDF URL feed - a set of URLs with context covering malware that infects IoT
(Internet of Things) devices
  Category : MaliciousHashDF
    Count of rules : 1
    Description : Kaspersky Lab MaliciousHashDF feed
                  Malicious Hash feed - a set of hashes of malicious objects
  Category : PhishingURLsDF
    Count of rules : 11167
    Description : Kaspersky Lab PhishingURLsDF feed
                  Phishing URL feed - a set of URLs with context that cover phishing websites and
web pages
```

**show security ips counters**

Данной командой выполняется просмотр счетчиков работы правил сервиса IPS/IDS.

**Синтаксис**

```
show security ips counters
```

**Необходимый уровень привилегий**

10

**Командный режим**

ROOT

**Пример**

```
esr# show security ips counters
-----
IPS general counters
-----
Packets decoded by ips engine:      83971
Invalid packets decoded by ips engine:  0
Packets accepted by ips engine:      83977
Packets blocked by ips engine:       0
Packets replaced by ips engine:       0
Alerts generated:                    8
-----
IPS Decoder engine
-----
Packets decoded by ips engine:      83971
Bytes decoded by ips engine:        125677543
Invalid packets decoded by ips engine:  0
IPv4 packets decoded by ips engine:   83971
IPv6 packets decoded by ips engine:    0
TCP packets decoded by ips engine:    75
UDP packets decoded by ips engine:    83891
SCTP packets decoded by ips engine:    0
ICMPv4 packets decoded by ips engine:  5
ICMPv6 packets decoded by ips engine:  0
PPP packets decoded by ips engine:    0
PPPoE packets decoded by ips engine:  0
GRE packets decoded by ips engine:    0
Teredo packets decoded by ips engine:  0
Average packets size decoded by ips engine: 1496
Maximum packets size decoded by ips engine: 1500
-----
IPS Application Layer
-----
HTTP Flow decoded by ips engine:      0
FTP Flow decoded by ips engine:        0
FTP-DATA Flow decoded by ips engine:    0
SMTP Flow decoded by ips engine:        0
TLS Flow decoded by ips engine:         0
SSH Flow decoded by ips engine:         0
IMAP Flow decoded by ips engine:        0
SMB Flow decoded by ips engine:         0
DCE/RPC flow over TCP decoded by ips engine: 0
```

```
DCE/RPC flow over UDP decoded by ips engine: 0
DNS flow over TCP decoded by ips engine:      0
DNS flow over UDP decoded by ips engine:      0
ENIP flow over TCP decoded by ips engine:     0
ENIP flow over UDP decoded by ips engine:     0
```

---

#### IPS Flow engine

---

```
TCP Flow decoded by ips engine:               1
UDP Flow decoded by ips engine:               1
ICMPv4 Flow decoded by ips engine:           1
ICMPv6 Flow decoded by ips engine:           0
Failed TCP Flow decoded by ips engine:        0
Failed UDP Flow decoded by ips engine:        1
```

---

#### IPS TCP engine

---

```
TCP sessions decoded by ips engine:          1
TCP SYN packets decoded by ips engine:        1
TCP SYN-ACK packets decoded by ips engine:    0
TCP RST packets decoded by ips engine:        0
TCP packets with invalid checksum:           0
TCP packets with wrong thread:               0
Packets with TCP header length too small:    0
TCP packets with invalid options:            0
```

## show security ips counters application-layer

Данной командой выполняется просмотр счетчиков работы правил сервиса IPS/IDS на уровне приложений.

### Синтаксис

```
show security ips counters application-layer
```

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример

```
esr# show security ips counters application-layer
-----
IPS Application Layer
-----
HTTP Flow decoded by ips engine:              0
FTP Flow decoded by ips engine:               0
FTP-DATA Flow decoded by ips engine:          0
SMTP Flow decoded by ips engine:              0
TLS Flow decoded by ips engine:               0
SSH Flow decoded by ips engine:               0
IMAP Flow decoded by ips engine:              0
SMB Flow decoded by ips engine:               0
DCE/RPC flow over TCP decoded by ips engine: 0
DCE/RPC flow over UDP decoded by ips engine: 0
```

```
DNS flow over TCP decoded by ips engine: 0
DNS flow over UDP decoded by ips engine: 0
ENIP flow over TCP decoded by ips engine: 0
ENIP flow over UDP decoded by ips engine: 0
```

### show security ips counters decoder

Данной командой выполняется просмотр счетчиков работы правил сервиса IPS/IDS декодера.

#### Синтаксис

```
show security ips counters decoder
```

#### Необходимый уровень привилегий

10

#### Командный режим

ROOT

#### Пример

```
esr# show security ips counters decoder
-----
IPS Decoder engine
-----
Packets decoded by ips engine:      83971
Bytes decoded by ips engine:        125677543
Invalid packets decoded by ips engine: 0
IPv4 packets decoded by ips engine:  83971
IPv6 packets decoded by ips engine:  0
TCP packets decoded by ips engine:   75
UDP packets decoded by ips engine:   83891
SCTP packets decoded by ips engine:  0
ICMPv4 packets decoded by ips engine: 5
ICMPv6 packets decoded by ips engine: 0
PPP packets decoded by ips engine:   0
PPPoE packets decoded by ips engine: 0
GRE packets decoded by ips engine:   0
Teredo packets decoded by ips engine: 0
Average packets size decoded by ips engine: 1496
Maximum packets size decoded by ips engine: 1500
```

### show security ips counters flow

Данной командой выполняется просмотр счетчиков работы правил сервиса IPS/IDS на потоках.

#### Синтаксис

```
show security ips counters flow
```

#### Необходимый уровень привилегий

10



**Командный режим**

ROOT

**Пример**

```

esr# show security ips counters flow
-----
IPS Flow engine
-----
TCP Flow decoded by ips engine:      1
UDP Flow decoded by ips engine:      1
ICMPv4 Flow decoded by ips engine:   1
ICMPv6 Flow decoded by ips engine:   0
Failed TCP Flow decoded by ips engine: 0
Failed UDP Flow decoded by ips engine: 1
-----

```

**show security ips counters general**

Данной командой выполняется просмотр глобальных счетчиков работы правил сервиса IPS/IDS.

**Синтаксис**

```
show security ips counters general
```

**Необходимый уровень привилегий**

10

**Командный режим**

ROOT

**Пример**

```

esr# show security ips counters general
-----
IPS general counters
-----
Packets decoded by ips engine:      83971
Invalid packets decoded by ips engine: 0
Packets accepted by ips engine:     83977
Packets blocked by ips engine:      0
Packets replaced by ips engine:     0
Alerts generated:                   8
-----

```

**show security ips counters tcp**

Данной командой выполняется просмотр счетчиков работы правил сервиса IPS/IDS для TCP-сессий.

**Синтаксис**

```
show security ips counters tcp
```

**Необходимый уровень привилегий**

10

**Командный режим**

ROOT

**Пример**

```

esr# show security ips counters tcp
-----
IPS TCP engine
-----
TCP sessions decoded by ips engine:      1
TCP SYN packets decoded by ips engine:    1
TCP SYN-ACK packets decoded by ips engine: 0
TCP RST packets decoded by ips engine:    0
TCP packets with invalid checksum:        0
TCP packets with wrong thread:           0
Packets with TCP header length too small: 0
TCP packets with invalid options:        0

```

**show security ips status**

Данной командой выполняется просмотр числа правил, используемых в текущий момент системой IPS/IDS.

**Синтаксис**

```
show security ips status [detailed]
```

**Параметры**

detailed – показывает расширенную информацию об используемых правилах.

**Необходимый уровень привилегий**

10

**Командный режим**

ROOT

**Пример**

```

esr# show security ips status

Rule files processed:      3
Rules successfully loaded: 21724
Rules failed:              0

esr# show security ips status detailed
Rule files processed:      3
Rules successfully loaded: 21724

```

```

Rules failed:          0
Rules processed:      21727
IP-only inspecting:   1
Payload inspecting:   3980
Application layer inspecting: 18951
Decoder event:        0

```

### show security ips user-server

Данной командой выполняется просмотр информации об обновлении правил IPS/IDS с пользовательских серверов обновлений.

#### Синтаксис

```
show security ips user-server [<WORD>]
```

#### Параметры

<WORD> – имя сервера, задается строкой от 1 до 64 символа.

#### Необходимый уровень привилегий

10

#### Командный режим

ROOT

#### Пример

```

esr# sh security ips user-server
Server name           Files MD5              Next update
-----
content-provider      93633ab9a73248ea50d58c25b1ac806c  2020-10-06 12:27:40
TH                    919f51bdf44052bfc0953362aef11c0d  2020-10-06 12:36:40
Traffic-ID            e5e2f6472a397227c0d96f5df430a207  2020-10-06 12:36:40
Aggressive            cfc3547b50f3f9fec366ba5a1e51cd1f  2020-10-06 12:36:40
JA3-Fingerprint      439aa6e57c66826b92337672937d505b  2020-10-05 16:51:40
C2-Botnet             39e118bd3884b3dc1df4ca3a03c05df1  2020-10-05 16:51:40
SSL-BlackList        1d9c969f25791b9ee8c8c0ab8449d849  2020-10-05 16:51:40
ET-Open              d53d92248a1f7cdc040d669a76cf27bc  2020-10-06 12:36:40

```

### update security ips content-provider rules

Данной командой инициируется принудительное обновление правил IPS/IDS, распространяемых по коммерческой лицензии.

Фактическое начало процедуры обновления правил происходит с некоторой задержкой после введения команды. Максимальная величина задержки составляет 5 минут.

#### Синтаксис

```
update security ips content-provider rules
```

**Необходимый уровень привилегий**

15

**Командный режим**

ROOT

**Пример**

```
esr# update security ips content-provider rules
```

**update security ips content-provider rules-info**

Данной командой инициируется принудительный запрос информации о категориях правил IPS/IDS, доступных по текущей коммерческой лицензии.

Фактическое начало процедуры обновления правил происходит с некоторой задержкой после введения команды. Максимальная величина задержки составляет 5 минут.

**Синтаксис**

```
update security ips content-provider rules-info
```

**Необходимый уровень привилегий**

15

**Командный режим**

ROOT

**Пример**

```
esr# update security ips content-provider rules-info
```

**update security ips user-server rules**

Данной командой инициируется принудительное обновление правил IPS/IDS с пользовательского сервера обновлений.

Фактическое начало процедуры обновления правил происходит с некоторой задержкой после введения команды. Максимальная величина задержки составляет 5 минут.

**Синтаксис**

```
update security ips user-server rules <WORD>
```

**Параметры**

<WORD> – имя сервера, задается строкой от 1 до 64 символа.

**Необходимый уровень привилегий**

15

## Командный режим

ROOT

### Пример

```
esr# update security ips user-server rules ET-Open
```

## Настройка политики IPS/IDS

### category

Данной командой определяется категория правил IPS/IDS определённого вендора, распространяемых по коммерческой лицензии, и осуществляется переход в режим настройки данной категории.

Использование отрицательной формы команды (no) удаляет сконфигурированную категорию из настроек сервиса IPS/IDS.

### Синтаксис

```
category <CATEGORY>
```

```
no category { <CATEGORY> | all }
```

### Параметры

<CATEGORY> – категория правил.

Список доступных категорий можно посмотреть в контекстной подсказке или командой:

```
show security ips content-provider rules-info
```

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IPS-VENDOR

### Пример

```
esr(config-ips-policy-vendor)# category MobileBotnetCAndCDF
```

### external network-group

Данной командой устанавливается профиль IP-адресов, которые сервис IPS/IDS будет считать ненадежными.

Сам профиль IP-адресов должен быть предварительно создан.

Использование отрицательной формы команды (no) удаляет сконфигурированный профиль из настроек сервиса IPS/IDS.

**Синтаксис**

```
external network-group <OBJ-GROUP-NETWORK-NAME>
no external network-group
```

**Параметры**

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IPS-POLICY

**Пример**

```
esr(config-ips-policy)# external network-group WAN
```

**protect network-group**

Данной командой устанавливается профиль IP-адресов, которые будет защищать сервис IPS/IDS.

Сам профиль IP-адресов должен быть предварительно создан.

Использование отрицательной формы команды (no) удаляет сконфигурированный профиль из настроек сервиса IPS/IDS.

**Синтаксис**

```
protect network-group <OBJ-GROUP-NETWORK-NAME>
no protect network-group
```

**Параметры**

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IPS-POLICY

**Пример**

```
esr(config-ips-policy)# protect network-group LAN
```

**rules**

Данной командой указывается количество правил, которое необходимо скачать для данной категории.

Использование отрицательной формы команды (no) удаляет назначенное действие.

Команда применима только для правил, распространяемых по коммерческой лицензии.

### Синтаксис

```
rules { all | count <COUNT> | percent <PERCENT> | recommended }
no rules all
```

### Параметры:

- all – данной командой указывается, что система IPS/IDS будет работать с полным набором правил данной категории;
- count <COUNT> – данной командой указывается действующее число правил данной категории, с которым будет работать система IPS/IDS:
  - <COUNT> – число правил. Минимальное значение 1, максимальное значение зависит от категории правил.
- percent <PERCENT> – данной командой указывается процентное соотношение от общего числа правил данной категории, с которым будет работать система IPS/IDS:
  - <PERCENT> – процент от общего числа правил.
- recommended – данной командой указывается, что система IPS/IDS будет использовать рекомендованное количество правил в данной категории. Рекомендованное количество составляет 42% от общего числа правил.

Максимальное число правил по категориям можно посмотреть в контекстной подсказке или командой:

```
show security ips content-provider rules-info
```

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IPS-VENDOR-CATEGORY

### Пример

```
esr(config-ips-vendor-category)# rules percent 25
```

### rules action

Данная команда используется для указания действия, которое должно быть применено для трафика, удовлетворяющего правилам данной категории.

Использование отрицательной формы команды (no) удаляет назначенное действие.

Команда применима только для правил, распространяемых по коммерческой лицензии.

### Синтаксис

```
rules action { alert | reject | pass | drop }
no rules action
```

### Параметры:

- alert – прохождение трафика разрешается, и сервис IPS/IDS генерирует сообщение;

- reject – прохождение трафика запрещается. Если это TCP-трафик, отправителю и получателю посылается пакет TCP-RESET, для остального типа трафика посылается пакет ICMP-ERROR. Сервис IPS/IDS генерирует сообщение;
- pass – прохождение трафика разрешается;
- drop – прохождение трафика запрещается, и сервис IPS/IDS генерирует сообщение.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IPS-VENDOR-CATEGORY

### Пример

```
esr(config-ips-vendor-category)# rules action drop
```

### security ips policy

Данной командой создается политика настроек сервиса IPS/IDS с определенным именем и осуществляется переход в режим конфигурирования политики.

Использование отрицательной формы команды (no) удаляет сконфигурированную политику настроек сервиса IPS/IDS.

### Синтаксис

```
[no] security ips policy <POLICY_NAME>
```

### Параметры

<POLICY\_NAME> – имя политики сервиса IPS/IDS, задается строкой до 31 символов.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

### Пример

```
esr(config)# security ips policy OFFICE
```

### vendor

Данной командой определяется вендор правил IPS/IDS, распространяемых по коммерческой лицензии, и осуществляется переход в режим настройки данного вендора.

Использование отрицательной формы команды (no) удаляет сконфигурированного вендора из настроек сервиса IPS/IDS.



**Синтаксис**

```
vendor <VENDOR>  
no vendor <CATEGORY>
```

**Параметры**

<VENDOR> – вендор правил.

Список доступных вендоров можно посмотреть в контекстной подсказке или командой:

```
show security ips content-provider rules-info
```

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IPS-POLICY

**Пример**

```
esr(config-ips-policy)# vendor kaspersky
```

**Настройка IPS****fail-close enable**

При использовании данной команды передача трафика через интерфейс, на котором настроен сервис IPS/IDS, в момент начальной загрузки маршрутизатора, будет заблокирована до тех пор, пока сервис IPS/IDS не запустится и не применит хотя бы одно сконфигурированное или существующее правило.

Использование отрицательной формы команды (no) разрешает прохождение трафика при отсутствии загруженных правил IPS/IDS.

**Синтаксис**

```
[no] fail-close enable
```

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IPS

**Пример**

```
esr(config-ips)# fail-close enable
```

## logging ips severity

Данной командой устанавливается уровень важности сообщения для логирования событий IPS/IDS. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
logging ips severity <SEVERITY>
no logging ips severity
```

### Параметры

<SEVERITY> – уровень важности сообщения, принимает значения (в порядке убывания важности):

- emerg – в системе произошла критическая ошибка, система неработоспособна;
- alert – сигналы тревоги, необходимо немедленное вмешательство персонала;
- crit – критическое состояние системы, сообщение о событии;
- error – сообщения об ошибках;
- warning – предупреждения, неаварийные сообщения;
- notice – сообщения о важных системных событиях;
- info – информационные сообщения системы;
- debug – отладочные сообщения, предоставляют пользователю информацию для корректной настройки системы;
- none – отключает вывод syslog-сообщений.

### Значение по умолчанию

info

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

### Пример

```
esr(config)# logging ips severity error
```

## logging remote-server

Данной командой задаются параметры удаленного сервера для отправки статистики работы сервиса IPS/IDS в формате EVE (elasticsearch).

Использование отрицательной формы команды (no) останавливает отправку статистики.

### Синтаксис

```
logging remote-server { <ADDR> | <IPV6-ADDR> } [ <TRANSPORT> ] [ <PORT> ] [ source-
address { <SRC-ADDR> | <IPV6-SRC-ADDR> | object-group <NETWORK_OBJ_GROUP_NAME> } ]
no logging remote-server
```

## Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPv6-ADDR> – IPv6-адрес, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<TRANSPORT> – протокол передачи данных, по умолчанию – UDP, принимает значения:

- TCP – передача данных осуществляется по протоколу TCP;
- UDP – передача данных осуществляется по протоколу UDP.

<PORT> – номер TCP/UDP-порта, опциональный параметр, принимает значения [1..65535], по умолчанию 514;

<SRC-ADDR> – IPv4-адрес маршрутизатора, который будет использоваться в качестве IP-адреса источника в отправляемых syslog-пакетах, по умолчанию – IPv4-адрес интерфейса, с которого отправляются пакеты;

<IPv6-SRC-ADDR> – IPv6-адрес маршрутизатора, который будет использоваться в качестве IPv6-адреса источника в отправляемых syslog-пакетах, по умолчанию – IPv6-адрес интерфейса, с которого отправляются пакеты;

<NETWORK\_OBJ\_GROUP\_NAME> – список адресов, которые будут использоваться в качестве source address.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IPS

## Пример

```
esr(config-ips)# logging remote-server 192.168.0.101
```

## logging update-interval

Данной командой устанавливается интервал отправки статистики работы сервиса IPS/IDS в формате EVE (elasticsearch).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
logging update-interval <INTERVAL>
```

```
no logging update-interval
```

## Параметры

<INTERVAL> – интервал отправки статистики работы сервиса IPS/IDS, задаётся в минутах.

## Значение по умолчанию

10 минут.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IPS

## Пример

```
esr(config-ips)# logging update-interval 10
```

## performance max

Данная команда позволяет сервису IPS/IDS использовать все ресурсы устройства для достижения максимальной производительности. Рекомендуется применять, когда устройство используется исключительно в качестве IPS/IDS. Не рекомендуется применять, когда помимо IPS/IDS устройство выполняет другие функции (маршрутизация, BRAS и т. д.).

Использование отрицательной формы команды (no) освобождает часть ресурсов устройства для использования другими сервисами.

## Синтаксис

```
[no] performance max
```

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IPS

## Пример

```
esr(config-ips)# performance max
```

## policy

Данной командой назначается созданная ранее политика настроек сервиса IPS/IDS.

Использование отрицательной формы команды (no) снимает назначенную ранее политику настроек сервиса IPS/IDS.

## Синтаксис

```
policy <POLICY_NAME>
```

```
no policy
```

## Параметры

<POLICY\_NAME> – имя политики сервиса IPS, задаётся строкой до 32 символов.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IPS

## Пример

```
esr(config-ips)# policy OFFICE
```

## queue-limit

Данной командой определяется предельное количество пакетов для виртуальной очереди одного потока сервиса IPS/IDS. Большой размер очереди позволяет оптимизировать производительность при всплесках трафика, но слишком большая очередь может привести к снижению производительности.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
queue-limit <QUEUE-LIMIT>
```

```
no queue-limit
```

## Параметры

<QUEUE-LIMIT> – предельное количество пакетов в виртуальной очереди, принимает значения в диапазоне [32..4096].

## Значение по умолчанию

1024

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IPS

## Пример

```
esr(config-ips)# queue-limit 2048
```

## default-packet-size follow-mtu

Данной командой включается вычисление максимального MTU среди интерфейсов, участвующих в работе IPS. При включении опции, значение default-packet-size становится равно вычисленному MTU + размер Ethernet-заголовка.

Использование отрицательной формы команды (no) сбрасывает значение MTU в конфигурации IPS/IDS на стандартное 1514 (размер MTU 1500 + заголовок ethernet 14).

### Синтаксис

```
default-packet-size follow-mtu  
[no] default-packet-size
```

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IPS

### Пример

```
esr(config-ips)# default-packet-size follow-mtu
```

### security ips

Команда используется для создания профиля сервиса IPS/IDS и перехода в режим его конфигурирования.

### Синтаксис

```
security ips
```

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

### Пример

```
esr(config)# security ips
```

### service-ips

Данная команда используется для включения сервиса IPS/IDS на сетевом интерфейсе.

Использование отрицательной формы команды (no) выключает сервис IPS/IDS на сетевом интерфейсе.

### Синтаксис

```
service-ips { inline | monitor }  
[no] service-ips
```

### Параметры

inline – режим работы сервиса IPS/IDS, когда устройство ставится в разрыв сети. В этом режиме сервис IPS/IDS может детектировать и блокировать трафик;

monitor – режим работы сервиса IPS/IDS, когда устройство подключается к SPAN/RSPAN-порту. В этом режиме возможно только детектировать трафик, т. е. сервис работает в режиме IDS.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

### Пример

```
esr(config-if-gi)# service-ips inline
```

## Настройка параметров кэширования записей контентной фильтрации

### content-filter

Данной командой осуществляется переход в режим конфигурирования параметров кэширования записей контентной фильтрации.

### Синтаксис

```
content-filter
```

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IPS

### Пример

```
esr(config-ips)# content-filter
```

### uri cache-size

Данная команда позволяет указать количество записей, которые будут храниться в памяти устройства.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
[no] uri cache-size <NUMBER>
```

### Параметры

<NUMBER> – значение размера кэша контент-фильтра в количестве записей. Принимает значение в диапазоне [1; 32768]. Значение по умолчанию – 32768.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IPS-CONTENT-FILTER

### Пример

```
esr(config-ips-content-filter)# uri cache-size 940
```

### uri reachable-interval

Данная команда позволяет настроить среднее время, в течение которого запись URI всё ещё действительна.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
[no] uri reachable-interval <DAYS>
```

### Параметры

<DAYS> – количество дней, в течение которых запись URI всё ещё действительна. Принимает значения [1; 365]. Значение по умолчанию – 7.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IPS-CONTENT-FILTER

### Пример

```
esr(config-ips-content-filter)# uri reachable-interval 30
```



## Настройка автообновления правил IPS/IDS, распространяемых по коммерческой лицензии

### content-provider

Данной командой осуществляется переход в режим конфигурирования источника обновлений правил, распространяемых по коммерческой лицензии.

#### Синтаксис

```
content-provider
```

#### Необходимый уровень привилегий

15

#### Командный режим

CONFIG

#### Пример

```
esr(config)# content-provider
```

### host address

Данной командой задаётся адрес сервера обновлений правил, распространяемых по коммерческой лицензии.

#### Синтаксис

```
host address { <ADDR> | <IPV6-ADDR> | <WORD> }
```

#### Параметры

<ADDR> – IP-адрес сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес сервера, задаётся в виде X:X:X:X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<WORD> – DNS-имя сервера, задаётся строкой до 31 символа.

#### Необходимый уровень привилегий

15

#### Командный режим

CONFIG-CONTENT-PROVIDER

#### Пример

```
esr(config-content-provider)# host address edm.eltex-co.ru
```

## host port

Данной командой задаётся номер TCP-порта сервера обновлений правил, распространяемых по коммерческой лицензии.

Использование отрицательной формы команды (no) удаляет номер TCP-порта сервера обновлений правил, распространяемых по коммерческой лицензии.

### Синтаксис

```
host port <PORT>  
no host port
```

### Параметры

<PORT> – номер TCP-порта, принимает значения [1..65535].

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-CONTENT-PROVIDER

### Пример

```
esr(config-content-provider)# host port 8098
```

## location

Данной командой задаётся текстовое описание, которое передаётся на сервер EDM-Issue.

Использование отрицательной формы команды (no) удаляет это описание.

### Синтаксис

```
location <WORD>  
no location
```

### Параметры

<WORD> – описание, задаётся строкой до 255 символов.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-CONTENT-PROVIDER

### Пример

```
esr(config-content-provider)# location "Server room in Novokuznetsk office"
```

## reboot

Данной командой задаётся время перезагрузки устройства при получении системной лицензии. Устройство перезагружается при первом подключении к серверу обновлений правил, распространяемых по коммерческой лицензии.

При наличии функциональной лицензии IPS/IDS перезагрузки не происходит.

### Синтаксис

```
reboot { immediately | time <TIME> }
```

### Параметры

`immediately` – перезагружаться сразу после получения лицензии;

`time <TIME>` – перезагружаться в указанное время `<TIME>`;

`<TIME>` – время перезагрузки в формате HH:MM:SS.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-CONTENT-PROVIDER

### Пример

```
esr(config-content-provider)# reboot time 05:00:00
```

## storage-path

Данной командой задается путь на внешнем носителе, по которому будут сохраняться правила IPS/IDS в зашифрованном виде, распространяемые по коммерческой лицензии.

Для использования с системой IPS/IDS на внешнем носителе должен быть создан раздел файловой системы в формате exFAT.

Использование отрицательной формы команды (`no`) останавливает сохранение правил.

### Синтаксис

```
storage-path { usb://<USB-NAME>:/ | mmc://<MMC-NAME>:/ }
```

```
no storage-device
```

### Параметры

`<USB-NAME>` – имя подключенного USB-носителя. Имя можно узнать в выводе команды [show storage-devices usb](#);

`<MMC-NAME>` – имя подключенного MMC-носителя. Имя можно узнать в выводе команды [show storage-devices mmc](#).

### Необходимый уровень привилегий config-ips-upgrade-user-server

15

## Командный режим

CONFIG-CONTENT-PROVIDER

### Пример

```
esr(config-content-provider)# storage-device usb://DATA/IPS
```

### system-name

Данной командой задаётся текстовое имя устройства, которое передаётся на сервер EDM-Issue.

Использование отрицательной формы команды (no) присваивает этому имени значение по умолчанию.

### Синтаксис

```
system-name <WORD>
```

```
no system-name
```

### Параметры

<WORD> – имя, задаётся строкой до 253 символа.

### Значение по умолчанию

По умолчанию значение system-name совпадает с hostname.

### Необходимый уровень привилегий

15

## Командный режим

CONFIG-CONTENT-PROVIDER

### Пример

```
esr(config-content-provider)# system-name main-office
```

### upgrade interval

Команда задаёт частоту, с которой устройство будет проверять обновления правил IPS/IDS, распространяемых по коммерческой лицензии.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
upgrade interval <HOURS>
```

```
no upgrade interval
```

### Параметры

<HOURS> – интервал обновлений в часах, от 1 до 240.

**Значение по умолчанию**

24

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-CONTENT-PROVIDER

**Пример**

```
esr(config-content-provider)# upgrade interval 36
```

**Настройка автообновления правил IPS/IDS из внешних источников****auto-upgrade**

Данной командой осуществляется переход в режим конфигурирования источников обновлений правил для сервиса.

**Синтаксис**

auto-upgrade

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IPS

**Пример**

```
esr(config-ips)# auto-upgrade
```

**storage-path**

Данной командой задается путь на внешнем носителе, по которому будут сохраняться правила IPS/IDS, загружаемые с пользовательских серверов обновлений.

Для использования с системой IPS/IDS на внешнем носителе должен быть создан раздел файловой системы в формате exFAT.

Использование отрицательной формы команды (no) останавливает сохранение правил.

**Синтаксис**

```
storage-path { usb://<USB-NAME>:/ | mmc://<MMC-NAME>:/ }  
no storage-device
```

## Параметры

<USB-NAME> – имя подключенного USB-носителя. Имя можно узнать в выводе команды [show storage-devices usb](#);

<MMC-NAME> – имя подключенного MMC-носителя. Имя можно узнать в выводе команды [show storage-devices mmc](#).

## Необходимый уровень привилегий config-ips-upgrade-user-server

15

## Командный режим

CONFIG-IPS

## Пример

```
esr(config-ips)# storage-device usb://DATA/
```

## upgrade interval

Команда задаёт частоту, с которой устройство будет проверять обновления правил IPS/IDS и/или файла классификатора IPD/IDS для данного URL.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
upgrade interval <HOURS>
```

```
no upgrade interval
```

## Параметры

<HOURS> – интервал обновлений в часах, от 1 до 240.

## Значение по умолчанию

24

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IPS-UPGRADE-USER-SERVER

## Пример

```
esr(config-ips-upgrade-user-server)# upgrade interval 36
```

## url

Команда используется для задания URL-ссылки.

Использование отрицательной формы команды (no) удаляет ссылку из конфигурации источника обновлений правил IPS/IDS.

### Синтаксис

```
url <URL>
```

```
no url
```

### Параметры

<URL> – текстовое поле, содержащее URL-ссылку длиной от 8 до 255 символов.

В качестве URL-ссылки может быть указан:

- файл правил с расширением .rule,
- файл классификатора правил с именем classification.config,
- каталог на сервере, содержащий файлы правил и/или файл классификатора правил.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IPS-UPGRADE-USER-SERVER

### Пример

```
esr(config-ips-upgrade-user-server)# url https://rules.emergingthreats.net/open/suricata-4.0/rules/
```

### user-server

Данной командой задаётся имя пользовательского сервера обновлений правил IPS/IDS и осуществляется переход в режим конфигурирования параметров пользовательского сервера обновлений.

Использование отрицательной формы команды (no) удаляет пользовательский сервер обновлений правил IPS/IDS и все правила, полученные с этого сервера.

### Синтаксис

```
user-server <WORD>
```

```
no user-server { <WORD> | all }
```

### Параметры

<WORD> – имя сервера, задается строкой от 1 до 64 символа.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IPS-AUTO-UPGRADE

## Пример

```
esr(config-ips-auto-upgrade)# user-server ET-Open
```

## Настройка пользовательских правил IPS/IDS

### action

Данная команда используется для указания действия, которое должно быть применено для трафика, удовлетворяющего заданным критериям.

Использование отрицательной формы команды (no) удаляет назначенное действие.

### Синтаксис

```
action { alert | reject | pass | drop }
```

```
no action
```

### Параметры

- alert – прохождение трафика разрешается, и сервис IPS/IDS генерирует сообщение;
- reject – прохождение трафика запрещается. Если это TCP-трафик, отправителю и получателю посылаются пакеты TCP-RESET, для остального типа трафика посылаются пакеты ICMP-ERROR. Сервис IPS/IDS генерирует сообщение;
- pass – прохождение трафика разрешается;
- drop – прохождение трафика запрещается, и сервис IPS/IDS генерирует сообщение.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IPS-CATEGORY-RULE

## Пример

```
esr(config-ips-category-rule)# action reject
```

### destination-address

Данной командой устанавливаются IP-адреса получателя, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
destination-address { ip <ADDR> | ip-prefix <ADDR/LEN> | object-group <OBJ_GR_NAME> |  
policy-object-group { protect | external } | any }
```

```
no destination-address
```



## Параметры

<ADDR> – IP-адрес получателя, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<ADDR/LEN> – IP-подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32];

<OBJ\_GR\_NAME> – имя профиля IP-адресов, который содержит IP-адреса назначения, задаётся строкой до 31 символа;

destination-address policy-object-group protect – устанавливает в качестве адресов назначения protect-адреса, определенные в политике IPS/IDS;

destination-address policy-object-group external – устанавливает в качестве адресов назначения external-адреса, определенные в политике IPS/IDS;

При указании значения «any» правило будет срабатывать для любого IP-адреса получателя.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IPS-CATEGORY-RULE

## Пример

```
esr(config-ips-category-rule)# destination-address ip 10.10.10.1
```

## destination-port

Данной командой устанавливается номер TCP/UDP-порта получателя, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) удаляет назначение.

**Синтаксис**

```
destination-port { any | <PORT> | object-group <OBJ-GR-NAME> }
no destination-port
```

**Параметры**

<PORT> – номер TCP/UDP-порта получателя, принимает значения [1..65535];

<OBJ\_GR\_NAME> – имя профиля TCP/UDP-портов получателя, задаётся строкой до 31 символа.

При указании значения «any» правило будет срабатывать для любого TCP/UDP-порта получателя.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IPS-CATEGORY-RULE

**Пример**

```
esr(config-ips-category-rule)# destination-port 22
```

**direction**

Данной командой устанавливается направление потока трафика, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) удаляет назначение.

**Синтаксис**

```
direction { one-way | round-trip }
no direction
```

**Параметры**

- one-way – трафик передаётся в одну сторону;
- round-trip – трафик передаётся в обе стороны.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IPS-CATEGORY-RULE

**Пример**

```
esr(config-ips-category-rule)# direction one-way
```

## ip dscp

Данной командой устанавливается значение кода DSCP, трафик которого будет обрабатываться в данном правиле.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
ip dscp <DSCP>
```

```
[no] ip dscp
```

### Параметры

<DSCP> – значение кода DSCP, принимает значения [0..63].

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IPS-CATEGORY-RULE

### Пример

```
esr(config-ips-category-rule)# ip dscp 8
```

## ip ftp command

Данной командой устанавливаются значения ключевых слов протокола FTP, для которых должно срабатывать правило.

Данная команда применима только для значения protocol ftp.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
ip ftp command <COMMAND>
```

```
[no] ip ftp command
```

### Параметры

<COMMAND> – может принимать следующие значения:

- <retr> – скачать файл;
- <stor> – залить файл;
- <mkd> – создать директорию;
- <rmr> – удалить директорию;
- <appe> – добавить в конец файла (с созданием);
- <dele> – удалить файл.

### Необходимый уровень привилегий

15

## Командный режим

### CONFIG-IPS-CATEGORY-RULE

#### Пример

```
esr(config-ips-category-rule)# protocol ftp
esr(config-ips-category-rule)# ip ftp command allo
```

#### ip ftp-data command

Данной командой устанавливаются значения ключевых слов протокола FTP-DATA, для которых должно срабатывать правило.

Данная команда применима только для значения protocol ftp-data.

Использование отрицательной формы команды (no) отменяет назначение.

#### Синтаксис

```
ip ftp-data command <COMMAND>
```

```
[no] ip ftp-data command
```

#### Параметры

<COMMAND> – может принимать следующие значения:

- <retr> – скачать файл;
- <stor> – залить файл;
- <appe> – добавить в конец файла (с созданием).

#### Необходимый уровень привилегий

15

## Командный режим

### CONFIG-IPS-CATEGORY-RULE

#### Пример

```
esr(config-ips-category-rule)# protocol ftp-data
esr(config-ips-category-rule)# ip ftp-data command stor
```

#### ip http

Данной командой устанавливаются значения ключевых слов протокола HTTP, для которых должно срабатывать правило.

Данная команда применима только для значения protocol http.

Использование отрицательной формы команды (no) отменяет назначение.

#### Синтаксис

```
ip http <COMMAND>
```

[no] ip http

## Параметры

<COMMAND> – может принимать следующие значения:

- accept;
- accept-enc;
- accept-lang;
- client-body;
- connection;
- content-len;
- content-type;
- cookie;
- file-data;
- header;
- header-names;
- host;
- protocol;
- referer;
- request-line;
- response-line;
- server-body;
- start;
- stat-code;
- stat-msg;
- uri;
- urilen <VALUE>;
- urilen comparison-operator { greater-than | less-than};
- user-agent.

Значения и применение ключевых слов протокола HTTP подробно описаны в документации SNORT 2.X/Suricata 4.X.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IPS-CATEGORY-RULE

## Пример

```
esr(config-ips-category-rule)# payload content "HTTP/1.0"
esr(config-ips-category-rule)# ip http protocol
```

## ip http content-filter

Данной командой назначается профиль категорий контентной фильтрации. Текущее правило будет срабатывать для http-сайтов, которые относятся к категориям заданным в этом профиле.

Сам профиль контентной фильтрации должен быть предварительно создан.

Данная команда применима только для значения protocol http.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
ip http content-filter <NAME>
[no] ip http content-filter
```

## Параметры

<NAME> – имя профиля контентной фильтрации, задаётся строкой до 31 символа.  
any – правило будет срабатывать для http-сайтов любой категории.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IPS-CATEGORY-RULE

## Пример

```
esr(config-ips-category-rule)# ip http content-filter Black-List
```

## ip http method

Данной командой устанавливаются значения метода доступа по протоколу HTTP, для которых должно срабатывать правило.

Данная команда применима только для значения protocol http.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
ip http method <COMMAND>
[no] ip http method
```

## Параметры

<COMMAND> – может принимать следующие значения:

- <GET> – запрашивает представление ресурса. Запросы с использованием этого метода могут только извлекать данные;
- <HEAD> – запрашивает ресурс так же, как и метод GET, но без тела ответа;
- <POST> – используется для отправки сущностей к определённому ресурсу;
- <PUT> – заменяет все текущие представления ресурса данными запроса;
- <DELETE> – удаляет указанный ресурс;
- <CONNECT> – устанавливает «туннель» к серверу, определённому по ресурсу;
- <OPTIONS> – используется для описания параметров соединения с ресурсом;
- <TRACE> – выполняет вызов возвращаемого тестового сообщения с ресурса;
- <PATCH> – используется для частичного изменения ресурса.

## Необходимый уровень привилегий

15

## Командный режим

### CONFIG-IPS-CATEGORY-RULE

#### Пример

```
esr(config-ips-category-rule)# ip http method get
```

#### **ip icmp code**

Данной командой устанавливается значение ICMP CODE, при котором сработает правило.

Данная команда применима только для значения protocol icmp.

Использование отрицательной формы команды (no) отменяет назначение.

#### Синтаксис

```
ip icmp code <CODE>
```

```
[no] ip icmp code
```

#### Параметры

<CODE> – значение CODE протокола ICMP, принимает значение в диапазоне [0..255].

#### Необходимый уровень привилегий

15

## Командный режим

### CONFIG-IPS-CATEGORY-RULE

#### Пример

```
esr(config-ips-category-rule)# ip icmp code 5
```

#### **ip icmp code comparison-operator**

Оператор сравнения для команды ip icmp code. Применим только совместно с этой командой.

Использование отрицательной формы команды (no) отменяет сравнение.

#### Синтаксис

```
ip icmp code comparison-operator { greater-than | less-than }
```

```
[no] ip icmp code comparison-operator
```

#### Параметры

- greater-than – больше чем;
- less-than – меньше чем.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IPS-CATEGORY-RULE

## Пример

```
esr(config-ips-category-rule)# ip icmp code 5
esr(config-ips-category-rule)# ip icmp code comparison-operator less-than
```

## ip icmp id

Данной командой устанавливается значение ICMP ID, при котором сработает правило.

Данная команда применима только для значения protocol icmp.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
ip icmp id <ID>
```

```
[no] ip icmp id
```

## Параметры

<ID> – значение ID протокола ICMP, принимает значение в диапазоне [0..65535].

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IPS-CATEGORY-RULE

## Пример

```
esr(config-ips-category-rule)# ip icmp id 65000
```

## ip icmp sequence-id

Данной командой устанавливается значение ICMP sequence-ID, при котором сработает правило.

Данная команда применима только для значения protocol icmp.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
ip icmp sequence-id <SEQ-ID>
```

```
[no] ip icmp sequence-id
```



**Параметры**

<SEQ-ID> – значение Sequence-ID протокола ICMP, принимает значение в диапазоне [0..4294967295].

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IPS-CATEGORY-RULE

**Пример**

```
esr(config-ips-category-rule)# ip icmp sequence-id 8388608
```

**ip icmp type**

Данной командой устанавливается значение ICMP TYPE, при котором сработает правило.

Данная команда применима только для значения protocol icmp.

Использование отрицательной формы команды (no) отменяет назначение.

**Синтаксис**

```
ip icmp type <TYPE>
```

```
[no] ip icmp type
```

**Параметры**

<TYPE> – значение TYPE протокола ICMP, принимает значение в диапазоне [0..255].

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IPS-CATEGORY-RULE

**Пример**

```
esr(config-ips-category-rule)# ip icmp type 12
```

**ip icmp type comparison-operator**

Оператор сравнения для команды ip icmp type. Применим только совместно с этой командой.

Использование отрицательной формы команды (no) отменяет сравнение.

**Синтаксис**

```
ip icmp type comparison-operator { greater-than | less-than }
```

```
[no] ip icmp type comparison-operator
```

## Параметры

- greater-than – больше чем;
- less-than – меньше чем.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IPS-CATEGORY-RULE

## Пример

```
esr(config-ips-category-rule)# ip icmp type 14
esr(config-ips-category-rule)# ip icmp code comparison-operator greater-than
```

## ip protocol-id

Данной командой устанавливается идентификационный номер IP-протокола, трафик которого будет обрабатываться в данном правиле.

Данная команда применима только для значения protocol any.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
ip protocol-id <ID>
```

```
[no] ip protocol-id
```

## Параметры

<ID> – идентификационный номер IP-протокола [1..255].

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IPS-CATEGORY-RULE

## Пример

```
esr(config-ips-category-rule)# ip protocol-id 250
```

## ip tcp acknowledgment-number

Данной командой устанавливается значение TCP Acknowledgment-Number, при котором сработает правило.

Данная команда применима только для значения protocol tcp.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
ip tcp acknowledgment-number <ACK-NUM>  
[no] ip tcp acknowledgment-number
```

### Параметры

<ACK-NUM> – значение Acknowledgment-Number протокола TCP, принимает значение в диапазоне [0..4294967295].

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IPS-CATEGORY-RULE

### Пример

```
esr(config-ips-category-rule)# ip tcp acknowledgment-number 32
```

### ip tcp sequence-id

Данной командой устанавливается значение TCP Sequence-ID, при котором сработает правило.

Данная команда применима только для значения protocol tcp.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
ip tcp sequence-id <SEQ-ID>  
[no] ip tcp sequence-id
```

### Параметры

<SEQ-ID> – значение Sequence-ID протокола TCP, принимает значение в диапазоне [0..4294967295].

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IPS-CATEGORY-RULE

### Пример

```
esr(config-ips-category-rule)# ip tcp sequence-id 2542
```

**ip tcp window-size**

Данной командой устанавливается значение TCP Window Size, при котором сработает правило.

Данная команда применима только для значения protocol tcp.

Использование отрицательной формы команды (no) отменяет назначение.

**Синтаксис**

```
ip tcp window-size <SIZE>
```

```
[no] ip tcp window-size
```

**Параметры**

<SIZE> – значение Window-Size протокола TCP, принимает значение в диапазоне [1..65535].

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IPS-CATEGORY-RULE

**Пример**

```
esr(config-ips-category-rule)# ip tcp window-size 50
```

**ip tls content-filter**

Данной командой назначается профиль категорий контентной фильтрации. Текущее правило будет срабатывать для https-сайтов, которые относятся к категориям заданным в этом профиле.

Сам профиль контентной фильтрации должен быть предварительно создан.

Данная команда применима только для значения protocol tls.

Использование отрицательной формы команды (no) отменяет назначение.

**Синтаксис**

```
ip tls content-filter <NAME>
```

```
[no] ip tls content-filter
```

**Параметры**

<NAME> – имя профиля контентной фильтрации, задаётся строкой до 31 символа.

any – правило будет срабатывать для http-сайтов любой категории.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IPS-CATEGORY-RULE

## Пример

```
esr(config-ips-category-rule)# ip tls content-filter Black-List
```

## ip ttl

Данной командой устанавливается значение времени жизни IP-пакета, трафик которого будет обрабатываться в данном правиле.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
ip ttl <TTL>
```

```
[no] ip ttl
```

### Параметры

<TTL> – время жизни IP-пакета, принимает значения [1..255].

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IPS-CATEGORY-RULE

## Пример

```
esr(config-ips-category-rule)# ip ttl 8
```

## ip ttl comparison-operator

Оператор сравнения для команды ip ttl. Применим только совместно с этой командой.

Использование отрицательной формы команды (no) отменяет сравнение.

### Синтаксис

```
ip ttl comparison-operator { greater-than | less-than }
```

```
[no] ip ttl comparison-operator
```

### Параметры

- greater-than – больше чем;
- less-than – меньше чем.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IPS-CATEGORY-RULE

## Пример

```
esr(config-ips-category-rule)# ip ttl 5
esr(config-ips-category-rule)# ip ttl comparison-operator less-than
```

## meta classification-type

Данная команда определяет классификацию события, которое сгенерирует сервис IPS/IDS, когда сработает правило.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
meta classification-type { not-suspicious | unknown | bad-unknown | attempted-recon |
successful-recon-limited | successful-recon-largescale | attempted-dos | successful-dos |
attempted-user | unsuccessful-user | successful-user | attempted-admin | successful-admin
| rpc-portmap-decode | shellcode-detect | string-detect | suspicious-filename-detect |
suspicious-login | system-call-detect | tcp-connection | trojan-activity | unusual-
client-port-connection | network-scan | denial-of-service | non-standard-protocol |
protocol-command-decode | web-application-activity | web-application-attack | misc-
activity | misc-attack | icmp-event | inappropriate-content | policy-violation | default-
login-attempt }
```

```
[no] meta classification-type
```

## Параметры

- not-suspicious – неподозрительный трафик;
- unknown – неизвестный трафик;
- bad-unknown – потенциально плохой трафик;
- attempted-recon – попытка утечки информации;
- successful-recon-limited – утечка информации;
- successful-recon-largescale – масштабная утечка информации;
- attempted-dos – попытка отказа в обслуживании;
- successful-dos – отказ в обслуживании;
- attempted-user – попытка получения привилегий пользователя;
- unsuccessful-user – безуспешная попытка получения привилегий пользователя;
- successful-user – успешная попытка получения привилегий пользователя;
- attempted-admin – попытка получения привилегий администратора;
- successful-admin – успешная попытка получения привилегий администратора;
- rpc-portmap-decode – декодирование запроса RPC;
- shellcode-detect – обнаружен исполняемый код;
- string-detect – обнаружена подозрительная строка;
- suspicious-filename-detect – было обнаружено подозрительное имя файла;
- suspicious-login – была обнаружена попытка входа с использованием подозрительного имени пользователя;
- system-call-detect – обнаружен системный вызов;
- tcp-connection – обнаружено TCP-соединение;
- trojan-activity – был обнаружен сетевой троян;
- unusual-client-port-connection – клиент использовал необычный порт;
- network-scan – обнаружение сетевого сканирования;
- denial-of-service – обнаружение атаки отказа в обслуживании;
- non-standard-protocol – обнаружение нестандартного протокола или события;
- protocol-command-decode – обнаружена попытка шифрования;
- web-application-activity – доступ к потенциально уязвимому веб-приложению;
- web-application-attack – атака на веб-приложение;

- misc-activity – прочая активность;
- misc-attack – прочие атаки;
- icmp-event – общее событие ICMP;
- inappropriate-content – обнаружено неприемлемое содержание;
- policy-violation – потенциальное нарушение корпоративной конфиденциальности;
- default-login-attempt – попытка входа с помощью стандартного логина/пароля.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IPS-CATEGORY-RULE

### Пример

```
esr(config-ips-category-rule)# meta classification-type misc-attack
```

### meta log-message

Данная команда определяет текстовое сообщение, которое сгенерирует сервис IPS/IDS, когда сработает правило.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
meta log-message <MESSAGE>
```

```
[no] meta log-message
```

### Параметры

<MESSAGE> – текстовое сообщение, задаётся строкой до 128 символов.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IPS-CATEGORY-RULE

### Пример

```
esr(config-ips-category-rule)# meta log-message "Possible SlowLorys attack"
```

### payload content

Данной командой можно указать содержимое IP-пакетов, при совпадении с которым будет срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

**Синтаксис**

```
payload content <CONTENT>
[no] payload content <CONTENT>
```

**Параметры**

<CONTENT> – текстовое сообщение, задаётся строкой до 1024 символов.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IPS-CATEGORY-RULE

**Пример**

```
esr(config-ips-category-rule)# payload content "virus"
```

**payload data-size**

Данной командой устанавливается размер содержимого пакетов, при котором сработает правило. Использование отрицательной формы команды (no) отменяет назначение.

**Синтаксис**

```
payload data-size <SIZE>
[no] payload data-size
```

**Параметры**

<SIZE> – размер содержимого пакетов, принимает значение в диапазоне [1..65535].

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IPS-CATEGORY-RULE

**Пример**

```
esr(config-ips-category-rule)# payload data-size 1024
```

**payload data-size comparison-operator**

Оператор сравнения для команды ip icmp type. Применим только совместно с этой командой. Использование отрицательной формы команды (no) отменяет сравнение.



**Синтаксис**

```
payload data-size comparison-operator { greater-than | less-than }
[no] payload data-size comparison-operator
```

**Параметры**

- greater-than – больше чем;
- less-than – меньше чем.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IPS-CATEGORY-RULE

**Пример**

```
esr(config-ips-category-rule)# payload data-size 1024
esr(config-ips-category-rule)# payload data-size comparison-operator less-than
```

**payload depth**

Данная команда указывает, сколько байтов с начала содержимого пакета будет проверять это правило. Команда используется только совместно с командой payload content. Может применяться совместно с командой payload offset.

Использование отрицательной формы команды (no) означает что будет проверяться все содержимое пакета на точное соответствие.

**Синтаксис**

```
payload depth <DEPTH>
[no] payload content depth
```

**Параметры**

<DEPTH> – число байт с начала содержимого пакета, принимает значение в диапазоне [1..65535].

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IPS-CATEGORY-RULE

**Пример**

```
esr(config-ips-category-rule)# payload content "abc"
esr(config-ips-category-rule)# payload depth 3
```

Под действие правила попадут пакеты с содержимым «abcdef», «abc123», «abcabcabc» и т.д.

## payload no-case

Данная команда указывает не различать прописные и заглавные буквы в описании содержимого пакетов. Команда используется только совместно с командой payload content.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
payload no-case
[no] payload content no-case
```

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IPS-CATEGORY-RULE

### Пример

```
esr(config-ips-category-rule)# payload content "virus"
esr(config-ips-category-rule)# payload no-case
```

Под действие правила попадут пакеты с содержимым «virus», «VIRUS», «ViRuS» и т.д.

## payload offset

Данная команда указывает число байт смещения от начала содержимого пакета, с которого начнется проверка. Команда используется только совместно с командой payload content. Может применяться совместно с командой payload depth.

Использование отрицательной формы команды (no) означает что будет проверяться все содержимое пакета на точное соответствие.

### Синтаксис

```
payload offset <OFFSET>
[no] payload content offset
```

### Параметры

<OFFSET> – число байт смещения от начала содержимого пакета, принимает значение в диапазоне [1..65535].

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IPS-CATEGORY-RULE

## Пример

```
esr(config-ips-category-rule)# payload content "abc"
esr(config-ips-category-rule)# payload depth 6
esr(config-ips-category-rule)# payload offset 3
```

Под действие правила попадут пакеты с содержимым «123abcdef», «defabc», «abcabcabc» и т. д.

## protocol

Данной командой устанавливается имя IP-протокола, для которого должно срабатывать правило. Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
protocol { any | ip | icmp | http | tcp | tls | udp }
[no] protocol
```

## Параметры

- any – правило будет срабатывать для любых протоколов;
- ip – правило сработает для протокола ip. В правиле можно настроить дополнительную фильтрацию командой ip protocol-id;
- icmp – правило сработает для протокола icmp. При выборе этого параметра значения source-port и destination-port должны быть any. В правиле можно настроить дополнительную фильтрацию командами ip icmp;
- http – правило сработает для протокола http. В правиле можно настроить дополнительную фильтрацию командами ip http;
- tcp – правило сработает для протокола tcp. В правиле можно настроить дополнительную фильтрацию командами ip tcp;
- tls – правило сработает для протокола https. В правиле можно настроить дополнительную фильтрацию командами ip tls;
- udp – правило сработает для протокола tcp. В правиле можно настроить дополнительную фильтрацию командами ip udp;
- ftp – правило сработает для протокола ftp. В правиле можно настроить дополнительную фильтрацию командами ip ftp;
- ftp-data – правило сработает для поля data протокола ftp. В правиле можно настроить дополнительную фильтрацию командами ip ftp-data.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IPS-CATEGORY-RULE

## Пример

```
esr(config-ips-category-rule)# protocol udp
```

## rule

Данная команда используется для создания правила и перехода в режим конфигурирования CONFIG-IPS-CATEGORY-RULE. Правила обрабатываются устройством в порядке возрастания их номеров.

Использование отрицательной формы команды (no) удаляет указанное правило.

### Синтаксис

```
[no] rule <ORDER>
```

### Параметры

<ORDER> – номер правила, принимает значения [1..512].

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IPS-CATEGORY

### Пример

```
esr(config-ips-category)# rule 10
esr(config-ips-category-rule)#
```

## security ips-category user-defined

Данной командой создается набор пользовательских правил сервиса IPS/IDS с определенным именем и осуществляется переход в режим конфигурирования этого набора.


Использование отрицательной формы команды (no) удаляет сконфигурированную политику настроек сервиса IPS.

### Синтаксис

```
[no] security ips-category user-defined <CATEGORY_NAME>
```

### Параметры

<CATEGORY\_NAME> – имя набора пользовательских правил сервиса IPS/IDS, задаётся строкой до 31 символов.

 В текущей версии использование символа «\_» (нижнее подчёркивание) в <CATEGORY\_NAME> недопустимо.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

## Пример

```
esr(config)# security ips-category user-defined PROTOCOL
esr(config-ips-category)#
```

## source-address

Данной командой устанавливаются IP-адреса отправителей, для которых должно срабатывать правило. Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
source-address {ip <ADDR> | ip-prefix <ADDR/LEN> | object-group <OBJ_GR_NAME> | policy-
object-group { protect | external } | any }
```

```
no source-address
```

## Параметры

<ADDR> – IP-адрес отправителя, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<ADDR/LEN> – IP-подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и LEN принимает значения [1..32];

<OBJ\_GR\_NAME> – имя профиля IP-адресов, который содержит IP-адреса отправителя, задаётся строкой до 31 символа;

destination-address policy-object-group protect – устанавливает в качестве адресов отправителя protect-адреса, определенные в политике IPS/IDS;

destination-address policy-object-group external – устанавливает в качестве адресов отправителя external-адреса, определенные в политике IPS/IDS.

При указании значения «any» правило будет срабатывать для любого IP-адреса отправителя.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IPS-CATEGORY-RULE

## Пример

```
esr(config-ips-category-rule)# source-address ip-prefix 192.168.0.0/16
```

## source-port

Данной командой устанавливается номер TCP/UDP-порта отправителя, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) удаляет назначение.

**Синтаксис**

```
source-port { any | <PORT> | object-group <OBJ-GR-NAME> }
no source-port
```

**Параметры**

<PORT> – номер TCP/UDP-порта отправителя, принимает значения [1..65535];

<OBJ\_GR\_NAME> – имя профиля TCP/UDP-портов отправителя, задаётся строкой до 31 символа.

При указании значения «any» правило будет срабатывать для любого TCP/UDP-порта отправителя.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IPS-CATEGORY-RULE

**Пример**

```
esr(config-ips-category-rule)# source-port 22
```

**threshold count**

Данной командой устанавливается пороговое значение количества пакетов, при котором сработает правило.

Использование отрицательной формы команды (no) удаляет назначение.

**Синтаксис**

```
threshold count <COUNT>
[no] threshold count
```

**Параметры**

<COUNT> – число пакетов, принимает значение в диапазоне [1..65535].

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IPS-CATEGORY-RULE

**Пример**

```
esr(config-ips-category-rule)# threshold count 1024
```

**threshold second**

Данной командой устанавливается интервал времени, для которого считается пороговое значение пакетов, при котором сработает правило. Команда используется только совместно с командой `threshold count`.

Использование отрицательной формы команды (`no`) удаляет назначение.

**Синтаксис**

```
threshold second <SECOND>
[no] threshold second
```

**Параметры**

<SECOND> – интервал времени в секундах, принимает значение в диапазоне [1..65535].

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IPS-CATEGORY-RULE

**Пример**

```
esr(config-ips-category-rule)# threshold second 1
```

**threshold track**

Данной командой устанавливается по адресу отправителя или получателя будут считаться пакеты, для которых устанавливаются пороговые значения. Команда используется только совместно с командой `threshold count`.

Использование отрицательной формы команды (`no`) удаляет назначение.

**Синтаксис**

```
threshold track { by_src | by_dst }
[no] threshold track
```

**Параметры**

- `by_src` – считать пороговое значение для пакетов с одинаковым IP отправителя;
- `by_dst` – считать пороговое значение для пакетов с одинаковым IP получателя.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-IPS-CATEGORY-RULE

## Пример

```
esr(config-ips-category-rule)# threshold track by-src
```

## threshold type

Данной командой устанавливается метод обработки пороговых значений. Команда используется только совместно с командой `threshold count`.

Использование отрицательной формы команды (`no`) удаляет назначение.

## Синтаксис

```
threshold type { treshhold | limit | both }
[no] threshold type
```

## Параметры

- `threshold` – выдавать сообщение каждый раз по достижении порога;
- `limit` – выдавать сообщение не чаще `<COUNT>` раз за интервал времени `<SECOND>`;
- `both` – комбинация `threshold` и `limit`. Сообщение будет генерироваться, если в течение интервала времени `<SECOND>` было `<COUNT>` или более пакетов, подходящих под условия правила, и сообщение будет отправлено только один раз в течении интервала времени.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IPS-CATEGORY-RULE

## Пример

```
esr(config-ips-category-rule)# threshold count 1024
esr(config-ips-category-rule)# threshold second 1
esr(config-ips-category-rule)# threshold track by-src
esr(config-ips-category-rule)# threshold type treshhold
```

Сообщение будет генерироваться на каждый X\*1025 пакет, приходящий за 1 секунду с одного IP-адреса.

## Настройка расширенных пользовательских правил

### rule-advanced

Данная команда используется для создания правила и перехода в режим конфигурирования CONFIG-IPS-CATEGORY-RULE-ADVANCED. Правила обрабатываются устройством в порядке возрастания их номеров.

Использование отрицательной формы команды (`no`) удаляет указанное правило.

## Синтаксис

```
[no] rule-advanced <ORDER>
```



## Параметры

<ORDER> – номер правила, принимает значения [1..4294967295].

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IPS-CATEGORY-RULE-ADVANCED

## Пример

```
esr(config-ips-category)# rule-advanced 10
esr(config-ips-category-rule-advanced)#
```

## rule-text

Данная команда описывает правило обработки трафика в формате SNORT 2.X/Suricata 4.X.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

rule-text <LINE>

[no] rule-text

## Параметры

<LINE> – текстовое сообщение в формате SNORT 2.X/Suricata 4.X, задаётся строкой до 1024 символов.

При написании правил в тексте правила необходимо использовать только двойные кавычки (символ "), а само правило необходимо заключать в одинарные кавычки (символ ').

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IPS-CATEGORY-RULE-ADVANCED

## Пример

```
esr(config-ips-category-rule-advanced)# rule-text 'alert tcp any any -> $HOME_NET any (msg:
"ATTACK [PTsecurity] Attempt to crash named using malformed RNDc packet"; flow: established,
to_server; content:"_auth"; depth: 20; fast_pattern; content: !"|02 00 00 00|"; within: 4;
content: "_ctrl"; content: "_ser"; content: "_tim"; content: "_exp"; reference: cve, 2016-1285;
classtype: attempted-dos; reference: url, github.com/ptresearch/AttackDetection; metadata: Open
Ptsecurity.com ruleset; sid: 10000005; rev: 3; )'
```

## 27 Управление QoS

- class
- class-map
- compression header ip tcp
- description
- fair-queue
- ip firewall sessions classification enable
- match access-group
- match application
- match dscp
- match url
- mode
- police
- policy-map
- priority class
- priority level
- priority-queue out
- priority-queue out num-of-queues
- qos dscp-mutation
- qos enable
- qos map cos-queue
- qos map dscp-mutation
- qos map dscp-queue
- qos queue default
- qos trust
- qos wrr-queue
- queue-limit
- random-detect
- random-detect queue
- random-detect queue default
- rate-limit
- service-policy
- service-policy
- set class-default cos
- set class-default dscp
- set class-default ip-precedence
- set cos
- set dscp
- set ip-precedence
- set queue
- shape auto-distribution
- shape average
- shape peak
- show qos interface shapers
- show qos map cos-queue
- show qos map dscp-mutation
- show qos map dscp-queue
- show qos policy binding
- show qos policy configuration
- show qos policy statistics
- show qos statistics
- show qos tunnel shapers
- traffic-shape

## class

Данная команда используется для привязки указанного QoS-класса к политике и осуществляется переход в режим настройки параметров класса.

Использование отрицательной формы команды (no) удаляет привязку класса к политике.

### Синтаксис

```
[no] class <NAME>
```

### Параметры

<NAME> – имя привязываемого класса, задается строкой до 31 символа. При указании значения «class-default» в данный класс попадает трафик неклассифицированный на входе.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-POLICY-MAP

### Пример

```
esr(config-policy-map)# class telnet_traffic
esr(config-class-policy-map)#
```

## class-map

Данной командой создается класс QoS и осуществляется переход в режим настройки параметров класса.

Использование отрицательной формы команды (no) удаляет созданный класс.

**i** Привила, описанные внутри класса, работают по принципу логического "ИЛИ".

Пример:

```
ESR(config)# class-map VOICE
ESR(config-class-map)# match dscp 12
ESR(config-class-map)# match dscp 56
ESR(config-class-map)# exit
```

Трафик, содержащий метки DSCP 12 или DSCP 56, будет попадать под критерий класса "VOICE".

### Синтаксис

```
[no] class-map <NAME>
```

### Параметры

<NAME> – имя создаваемого класса, задается строкой до 31 символа.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# class-map telnet_traffic
```

**compression header ip tcp**

Данной командой включается протокол компрессии tcp-заголовков для трафика отдельного класса.

Использование отрицательной формы команды (no) отменяет использование протокола компрессии tcp-заголовков для трафика отдельного класса.

**Синтаксис**

```
[no] compression header ip tcp
```

**Параметры**

Отсутствуют.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-CLASS-POLICY-MAP

**Пример**

```
esr(config-class-policy-map)# compression header ip tcp
```

**description**

Данной командой определяется описание классификатора QoS.

Использование отрицательной формы команды (no) удаляет описание.

**Синтаксис**

```
description <DESCRIPTION>
```

```
no description
```

## Параметры

<DESCRIPTION> – описание классификатора, задаётся строкой до 255 символов.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-CLASS-MAP

CONFIG-POLICY-MAP

## Пример

```
esr(config-class-map)# description "VoIP"
```

## fair-queue

Данной командой определяется предельное количество потоков для данного класса. Параметр характерен для очередей SFQ и FIFO.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
fair-queue <QUEUE-LIMIT>
```

```
no fair-queue
```

## Параметры

<QUEUE-LIMIT> – предельное количество виртуальных очередей, принимает значения в диапазоне [16..4096].

## Значение по умолчанию

16

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-CLASS-POLICY-MAP

## Пример

```
esr(config-class-policy-map)# fair-queue 200
```

## ip firewall sessions classification enable

Командой выполняется включение классифицирования сессий на основе политики QoS.

Использование отрицательной формы команды (no) отключает классифицирование сессий.

### Синтаксис

```
[no] ip firewall sessions classification enable
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# ip firewall sessions classification enable
```

## match access-group

Команда используется для привязки списка контроля доступа (ACL), по которому будет определяться отношение входящего трафика к конфигурируемому классу.

Использование отрицательной формы команды (no) удаляет привязку списка контроля доступа к данному классу.

### Синтаксис

```
[no] match access-group {<NAME> | all }
```

### Параметры

<NAME> – имя списка контроля доступа, задаётся строкой до 31 символа.

При удалении возможно использование ключа "all" для удаления всех заданных списков доступа.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-CLASS-MAP

### Пример

```
esr(config-class-map)# match access-group acl-ssh-traffic
```

### match application

Команда используется для привязки списка приложений, по которому будет определяться отношение входящего трафика к конфигурируемому классу.

Использование отрицательной формы команды (no) удаляет привязку списка приложений к данному классу.

❌ Классификация QoS по спискам приложений работает, только если firewall настроен на работы в StateLess-режиме.

### Синтаксис

```
[no] match application { <NAME> | all }
```

### Параметры

<NAME> – имя списка приложений, задаётся строкой до 31 символа.

При удалении возможно использование ключа "all" для удаления всех заданных списков доступа.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-CLASS-MAP

### Пример

```
esr(config-class-map)# match application APP-PERMIT
```

### match dscp

Данной командой устанавливается значение кода DSCP, трафик которого будет обрабатываться в данном class-map.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
[no] match dscp <DSCP>
```

## Параметры

<DSCP> – значение кода DSCP, принимает значения [0..63].

При удалении возможно использование ключа "all" для удаления всех заданных dscp.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-CLASS-MAP

## Пример

```
esr(config-class-map)# match dscp 55
```

## match url

Команда используется для привязки списка url, по которому будет определяться отношение входящего трафика к конфигурируемому классу.

Использование отрицательной формы команды (no) удаляет привязку списка url.

❌ Классификация QoS по спискам url работает, только если firewall настроен на работы в StateLess-режиме.

## Синтаксис

```
[no] match url { <NAME> | all }
```

## Параметры

<NAME> – имя списка url, задаётся строкой до 31 символа.

При удалении возможно использование ключа "all" для удаления всех заданных списков доступа.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-CLASS-MAP

## Пример

```
esr(config-class-map)# match url URL-PERMIT
```



## mode

Данной командой определяется режим работы класса.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
mode <MODE>
```

```
no mode
```

### Параметры

<MODE> – режим класса:

- fifo – режим FIFO (First In, First Out);
- gred – режим GRED (Generalized RED);
- red – режим RED (Random Early Detection);
- sfq – режим SFQ (очередь SFQ распределяет передачу пакетов на базе потоков).

### Значение по умолчанию

FIFO

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-CLASS-POLICY-MAP

### Пример

```
esr(config-class-policy-map)# mode red
```

## police

Данной командой настраивается прохождение, отбрасывание или маркировка пакетов, соответствующих конфигурируемому классу или политики в целом, исходя из заполненности токенами корзин conform и exceed.

Использование отрицательной формы команды (no) удаляет настроенный параметр.

### Синтаксис

```
police <RATE> [burst-conforming <BURST-CONFORM>] [conform-action <CONFORM-ACTION>]
[exceed-action <EXCEED-ACTION>] [burst-excess <BURST-EXCEED> [violate-action <VIOLATE-
ACTION>]]
```

```
no police
```

## Параметры

<RATE> – скорость пополнения токенами conform корзины, Кбит/с;

<BURST-CONFORM> – размер conform-корзины, Байт;

<BURST-EXCEED> – размер excess-корзины, Байт;

<VIOLATE-ACTION> – действие, которое необходимо выполнить с пакетом, для которого имеются токены conform-корзины, принимает значения { permit | deny | set-cos <COS> | set-dscp <DSCP> };

<CONFORM-ACTION> – действие, которое необходимо выполнить с пакетом, при исчерпании токенов conform-корзины, но имеются токены excess-корзины, принимает значения { permit | deny | set-cos <COS> | set-dscp <DSCP>};

<EXCEED-ACTION> – действие, которое необходимо выполнить с пакетом, для которого исчерпаны токены excess-корзины, принимает значения { permit | deny | set-cos <COS> | set-dscp <DSCP> };

<COS> – классификатор обслуживания в теге 802.1q пакета, принимает значения [0..7];

<DSCP> – значение кода DSCP, принимает значения [0..63].

## Значение по умолчанию

<RATE> – отсутствует;

<BURST-CONFORM> – 1500;

<BURST-EXCEED> – отсутствует;

<CONFORM-ACTION> – permit;

<EXCEED-ACTION> – deny;

<VIOLATE-ACTION> – deny;

<COS> – отсутствует;

<DSCP> – отсутствует.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-POLICY-MAP

CONFIG-CLASS-POLICY-MAP

## Пример

```
esr(config-policy-map)# police 256
```

```
esr(config-class-policy-map)# police 256 burst-excess 3000 exceed-action set-dscp 63
```

## policy-map

Данной командой создается политика QoS и осуществляется переход в режим настройки параметров политики.

Использование отрицательной формы команды (no) удаляет созданную политику.

### Синтаксис

```
[no] policy-map <NAME>
```

### Параметры

<NAME> – имя создаваемой политики, задается строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# policy-map input_policy
```

## priority class

Данной командой задается приоритет класса в WRR-процессе. Классы с наибольшим приоритетом обрабатываются в первую очередь.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
priority class <PRIORITY>
```

```
no priority
```

### Параметры

<PRIORITY> – приоритет класса в WRR-процессе, принимает значения [1..8].

### Значение по умолчанию

1

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-CLASS-POLICY-MAP

### Пример

```
esr(config-class-policy-map)# priority class 5
```

### priority level

Данной командой класс переводится в режим Strict Priority и задается приоритет класса. Классы с наибольшим приоритетом обрабатываются в первую очередь.

Использование отрицательной формы команды (no) переводит класс в режим WRR.

### Синтаксис

```
priority level <PRIORITY>
```

```
no priority
```

### Параметры

<PRIORITY> – приоритет класса в Strict Priority-процессе, принимает значения:

- для ESR-10/12V/12VF/15/15R/15VF/20/21/100/200/1000 – [1..8];
- для ESR-1200/1500/1511/1700 – [1..7].

### Значение по умолчанию

Класс работает в режиме WRR, приоритет не задан.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-CLASS-POLICY-MAP

### Пример

```
esr(config-class-policy-map)# priority level 5
```

### priority-queue out

Данная команда задаёт предельное количество пакетов в очереди.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
priority-queue out <QUEUE> limit <LIMIT>
```

```
no priority-queue out <QUEUE> limit
```

## Параметры

<QUEUE> – идентификатор очереди, принимает значение [1..8];

<LIMIT> – предельное количество пакетов в виртуальной очереди, принимает значения в диапазоне [100-1000].

## Значение по умолчанию

500

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI <sup>1</sup>

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL <sup>1</sup>

CONFIG-BRIDGE

CONFIG-LOOPBACK


CONFIG-IP4IP4

CONFIG-GRE

CONFIG-L2TPV3

## Пример

```
esr(config-if)# priority-queue out 1 limit 800
```

 <sup>1</sup> Данный функционал поддерживается только на маршрутизаторе ESR-10/12V/12VF/20/21/100/200.

## priority-queue out num-of-queues

Данная команда задает количество приоритетных очередей от восьмой к первой. Оставшиеся очереди являются взвешенными.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
priority-queue out num-of-queues <VALUE>
```

```
no priority-queue out num-of-queues
```

**Параметры**

<VALUE> – количество очередей, принимает значение [1..8], где:

- 0 – все очереди участвуют в WRR (WRR – механизм обработки очередей на основе веса);
- 8 – все очереди обслуживаются как «strict priority» (strict priority – приоритетная очередь обслуживается сразу, как только появляются пакеты).

**Значение по умолчанию**

8

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# priority-queue out num-of-queues 5
```

**qos dscp-mutation**

Данной командой включается применение изменений кодов DSCP в соответствии с таблицей DSCP-Mutation. Коды DSCP изменяются только для входящего трафика доверенных портов в режиме QoS Basic.

Использование отрицательной формы команды (no) отключает изменение кодов DSCP.

**Синтаксис**

```
[no] qos dscp-mutation
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

## Командный режим

CONFIG

### Пример

```
esr(config)# qos wrr-queue 3 bandwidth 130
```

### qos enable

Данной командой включается сервис QoS на интерфейсе. Если к интерфейсу не привязана политика QoS (привязка политик описана в разделе [service-policy](#)), то интерфейс работает в режиме Basic QoS, иначе Policy-based QoS.

Basic QoS – классификация трафика выполняется на основе кодов DSCP и/или 802.1p в зависимости от выбранного режима доверия (команда описана в разделе [qos trust](#)). Трафик направляется в очереди в соответствии с таблицами DSCP-Queue и/или CoS-Queue.

Policy-based QoS – классификация и направление трафика в очереди выполняется на основе QoS-политик. В каждой политике определяется набор классов, на которые необходимо разделить трафик. Отношение трафика к определенному классу политики определяется на входе в маршрутизатор правилами ACL (привязка ACL описана в разделе [match access-group](#)), для этого назначается QoS-политика на входящее направление. Для ограничения полосы ранее классифицированного трафика и других функций QoS-политика назначается на исходящее направление.

Использование отрицательной формы команды (no) выключает сервис QoS на интерфейсе.

### Синтаксис

```
[no] qos enable
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Выключено.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL  
CONFIG-CELLULAR-MODEM  
CONFIG-BRIDGE  
CONFIG-LOOPBACK  
CONFIG-IF-E1  
CONFIG-IP4IP4  
CONFIG-GRE  
CONFIG-SUBTUNNEL  
CONFIG-VTI  
CONFIG-L2TPV3  
CONFIG-PPPOE  
CONFIG-PPTP  
CONFIG-L2TP  
CONFIG-IF-MULTILINK  
CONFIG-OPENVPN

### Пример

```
esr(config-if-gi)# qos enable
```

### qos map cos-queue

Данная команда устанавливает соответствие между значениями кодов 802.1p входящих пакетов и исходящими очередями.

Использование отрицательной формы команды (no) устанавливает соответствие по умолчанию.

### Синтаксис

```
qos map cos-queue <COS> to <QUEUE>
```

```
no qos map dscp-queue <COS>
```

### Параметры

<COS> – классификатор обслуживания в теге 802.1p пакета, принимает значения [0..7];

<QUEUE> – идентификатор очереди, принимает значения [1..8].

### Значения по умолчанию:

CoS: (0), очередь 1

CoS: (1), очередь 2

CoS: (2), очередь 3

CoS: (3), очередь 4

CoS: (4), очередь 5

CoS: (5), очередь 6



CoS: (6), очередь 7

CoS: (7), очередь 8

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# qos map cos-queue 7 to 5
```

### qos map dscp-mutation

Данная команда устанавливает соответствие между значениями кодов DSCP входящих пакетов и кодов DSCP на выходе из устройства.

Использование отрицательной формы команды (no) устанавливает соответствие по умолчанию.

### Синтаксис

```
qos map dscp-mutation <DSCP> to <DSCP>
```

```
no qos map dscp-mutation <DSCP>
```

### Параметры

<DSCP> – классификатор обслуживания в IP-заголовке пакета, принимает значения [0..63].

### Значения по умолчанию:

Значения кодов DSCP входящих пакетов и кодов DSCP на выходе из устройства совпадают.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# qos map dscp-mutation 10 to 40
```

### qos map dscp-queue

Данная команда устанавливает соответствие между значениями кодов DSCP входящих пакетов и исходящими очередями.

Использование отрицательной формы команды (no) устанавливает соответствие по умолчанию.

### Синтаксис

```
qos map dscp-queue <DSCP> to <QUEUE>
```

```
no qos map dscp-queue <DSCP>
```

### Параметры

<DSCP> – классификатор обслуживания в IP-заголовке пакета, принимает значения [0..63];

<QUEUE> – внутренняя очередь, в которую необходимо отправлять пакеты с ранее описанным DSCP, принимает значение [1..8].

### Значения по умолчанию:

DSCP: (0-7), очередь 1

DSCP: (8-15), очередь 2

DSCP: (16-23), очередь 3

DSCP: (24-31), очередь 4

DSCP: (32-39), очередь 5

DSCP: (40-47), очередь 6

DSCP: (48-55), очередь 7

DSCP: (56-63), очередь 8

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# qos map dscp-queue 42 to 5
```

### qos queue default

Данная команда устанавливает номер очереди по умолчанию, в которую попадает весь трафик, кроме IP, в режиме доверия DSCP-приоритетам в случае с Basic QoS, а также неклассифицированный трафик в случае с Policy-based QoS.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
qos queue default <QUEUE>
```

```
no qos queue default
```

## Параметры

<QUEUE> – идентификатор очереди, принимает значения [1..8].

## Значение по умолчанию

1

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# qos queue default 3
```

## qos trust

Данной командой устанавливается режим доверия к значениям кодов 802.1p и DSCP во входящих пакетах для Basic QoS-режима работы интерфейса.

Использование отрицательной формы команды (no) устанавливает соответствие по умолчанию.

## Синтаксис

```
qos trust <MODE>
```

```
no qos trust
```

## Параметры

<MODE> – режим доверия к значениям кодов 802.1p и DSCP, принимает одно из следующих значений:

- dscp – режим доверия значениям кодов DSCP в IP-заголовке. Не IP-пакеты будут направлены в очередь по умолчанию (команда описана в разделе [qos queue default](#));
- cos – режим доверия значениям кодов 802.1p в теге 802.1q. Нетегированные пакеты будут направлены в очередь по умолчанию (команда описана в разделе [qos queue default](#));
- cos-dscp – режим доверия значениям кодов DSCP для IP-пакетов и значениям кодов 802.1p для остальных пакетов.

## Значения по умолчанию

Режим доверия значениям кодов DSCP (dscp).

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# qos trust cos-dscp
```

## qos wrr-queue

Данной командой определяются веса для соответствующих взвешенных очередей.

Использование отрицательной формы команды (no) устанавливает значение веса для указанной очереди по умолчанию.

## Синтаксис

```
qos wrr-queue <QUEUE> bandwidth <WEIGHT>
```

```
no qos wrr-queue <QUEUE>
```

## Параметры

<QUEUE> – идентификатор очереди, принимает значение [1..8];

<WEIGHT> – значение веса, принимает значение [1..255].

## Значение по умолчанию

1

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# qos wrr-queue 3 bandwidth 130
```

## queue-limit

Данной командой определяется предельное количество пакетов для виртуальной очереди.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
queue-limit <QUEUE-LIMIT>
```

```
no queue-limit
```

## Параметры

<QUEUE-LIMIT> – предельное количество пакетов в виртуальной очереди, принимает значения в диапазоне [2..4096].

## Значение по умолчанию

127

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-CLASS-POLICY-MAP

## Пример

```
esr(config-class-policy-map)# queue-limit 200
```

## random-detect

Данной командой определяются параметры RED.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
random-detect <LIMIT> <MIN> <MAX> <APS> <APS-NUM> <PROBABILITY>
```

```
no random-detect
```

## Параметры

<LIMIT> – предельный размер очереди в байтах, принимает значения в диапазоне [1..1000000];

<MIN> – минимальный размер очереди в байтах, принимает значения в диапазоне [1..1000000];

<MAX> – максимальный размер очереди в байтах, принимает значения в диапазоне [1..1000000];

<APS> – средний размер пакета в байтах, принимает значение в диапазоне [1..10000000];

<APS-NUM> – количество пакетов среднего размера, разрешенных для кратковременного пропуска, принимает значение в диапазоне [0..10000000];

<PROBABILITY> – вероятность отбрасывания пакетов, принимает значения [0..100].

При указании значений должны выполняться следующие правила:

<MAX> > 2 \* <MIN> <LIMIT> > 3 \* <MAX>

## Необходимый уровень привилегий

10

**Командный режим**

CONFIG-CLASS-POLICY-MAP

**Пример**

```
esr(config-class-policy-map)# random-detect 9000 1500 3000 1000 10 100
```

**random-detect queue**

Данной командой определяются параметры GRED.

Использование отрицательной формы команды (no) отменяет назначение.

**Синтаксис**

```
random-detect queue <QUEUE-NUM> [ dscp <DSCP> | precedence <IPP> ] <LIMIT> <MIN> <MAX>
<APS> <APS-NUM> <PROBABILITY>
```

```
no random-detect queue <PRECEDENCE>
```

**Параметры**

<QUEUE-NUM> – номер очереди [1..16];

<DSCP> – классификатор обслуживания в IP-заголовке пакета, принимает значения [0..63];

<IPP> – значение кода IP Precedence, принимает значения [0..7];

<PRECEDENCE> – значение IP Precedence [0..7];

<LIMIT> – предельный размер очереди в байтах, принимает значения в диапазоне [1..1000000];

<MIN> – минимальный размер очереди в байтах, принимает значения в диапазоне [1..1000000];

<MAX> – максимальный размер очереди в байтах, принимает значения в диапазоне [1..1000000];

<APS> – средний размер пакета в байтах, принимает значение в диапазоне [1..10000000];

<APS-NUM> – количество пакетов среднего размера, разрешенных для кратковременного пропуска, принимает значение в диапазоне [0..10000000];

<PROBABILITY> – вероятность отбрасывания пакетов, принимает значения [0..100].

При указании значений должны выполняться следующие правила:

<MAX> > 2 \* <MIN>

<LIMIT> > 3 \* <MAX>

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-CLASS-POLICY-MAP

## Пример

```
esr(config-class-policy-map)# random-detect queue 1 precedence 2 9000 1500 3000 1000 10 100
```

## random-detect queue default

Данной командой определяется очередь по умолчанию для механизма GRED.

**i** Для применения данной команды предварительно необходимо настроить команду:

```
random-detect queue <QUEUE-NUM> dscp <DSCP>/precedence <0-7> <LIMIT> <MIN> <MAX> <APS> <APS-  
NUM> <PROBABILITY>
```

После применения команд:

```
random-detect queue <QUEUE-NUM> dscp <DSCP>/precedence <0-7> <LIMIT> <MIN> <MAX> <APS> <APS-  
NUM> <PROBABILITY>
```

```
random-detect queue default <QUEUE-NUM>
```

поля dscp/precedence в заголовке IP-пакетов данной очереди будут игнорироваться.

Более подробно параметры указываемые в команде `random-detect queue` описаны в разделе [random-detect queue](#).

Использование отрицательной формы команды (`no`) отменяет назначение.

## Синтаксис

```
random-detect queue default <QUEUE-NUM>
```

```
no random-detect queue default <QUEUE-NUM>
```

## Параметры

<QUEUE-NUM> – номер очереди [1..16].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-CLASS-POLICY-MAP

## Пример

```
esr(config-class-policy-map)# random-detect queue default
```

## rate-limit

Данная команда устанавливает ограничение скорости входящего трафика. Команда актуальна только для Basic QoS режима интерфейса.

Использование отрицательной формы команды (`no`) снимает ограничение.

**Синтаксис**

```
rate-limit <BANDWIDTH> [BURST]  
no rate-limit
```

**Параметры**

<BANDWIDTH> – средняя скорость трафика в кбит/с, принимает значение [1..10000000] для TengigabitEthernet интерфейсов и [1..1000000] для прочих интерфейсов и туннелей;

<BURST> – размер сдерживающего порога в кБ, принимает значение [4..16000]. По умолчанию 128 кБ.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

```
CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-CELLULAR-MODEM  
CONFIG-BRIDGE  
CONFIG-LOOPBACK  
CONFIG-IF-E1  
CONFIG-IP4IP4  
CONFIG-GRE  
CONFIG-VTI  
CONFIG-L2TPV3  
CONFIG-L2TP  
CONFIG-PPPOE  
CONFIG-PPTP  
CONFIG-IF-MULTILINK  
CONFIG-OPENVPN
```



## Пример

```
esr(config-if-gi)# rate-limit 1670000
```

## service-policy

Данная команда используется для привязки указанной QoS-политики к конфигурируемому интерфейсу для классификации входящего (input) или приоритизации исходящего (output) трафика.

Использование отрицательной формы команды (no) удаляет привязку политики к данному интерфейсу.

## Синтаксис

```
service-policy { input | output } <NAME>
service-policy dynamic { upstream | downstream | all }
no service-policy { input | output }
```

## Параметры

<NAME> – имя QoS-политики, задаётся строкой до 31 символа;

dynamic – привязка динамической QoS-политики к конфигурируемому интерфейсу для приоритизации восходящего (upstream), нисходящего (downstream) или всего (all) трафика. Политика наполняется данными, полученными от RADIUS-сервера.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-CELLULAR-MODEM

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-IF-E1

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-VTI

CONFIG-L2TPV3

CONFIG-SUBTUNNEL (только с ключом "input")

CONFIG-PPPOE

CONFIG-PPTP

CONFIG-L2TP

CONFIG-IF-MULTILINK

CONFIG-OPENVPN

### Пример

```
esr(config-if-gi)# service-policy input input_policy
```

### service-policy

Данной командой привязывается политика QoS к классу для создания иерархического QoS.

Использование отрицательной формы команды (no) удаляет привязку политики к классу.

### Синтаксис

```
[no] service-policy <NAME>
```

### Параметры

<NAME> – имя политики, задается строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-CLASS-POLICY-MAP

### Пример

```
esr(config-class-policy-map)# service-policy input_policy
```

### set class-default cos

Данной командой задается значение 802.1p приоритета, которое будет установлено в пакетах, соответствующих классу по умолчанию (class-default).

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
set class-default cos <COS>
```

```
no set class-default cos
```

## Параметры

<COS> – классификатор обслуживания в теге 802.1q пакета, принимает значения [0..7].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-POLICY-MAP

## Пример

```
esr(config-policy-map)# set class-default cos 5
```

## set class-default dscp

Данной командой задается значение кода DSCP, которое будет установлено в IP-пакетах, соответствующих классу по умолчанию (class-default).

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
set class-default dscp <DSCP>
```

```
no set class-default dscp
```

## Параметры

<DSCP> – значение кода DSCP, принимает значения [0..63].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-POLICY-MAP

## Пример

```
esr(config-class-map)# set class-default dscp 16
```

## set class-default ip-precedence

Данной командой задается значение кода IP Precedence, которое будет установлено в IP-пакетах, соответствующих классу по умолчанию (class-default).

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
set class-default ip-precedence <IPP>  
no set class-default ip-precedence
```

### Параметры

<IPP> – значение кода IP Precedence, принимает значения [0..7].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-POLICY-MAP

### Пример

```
esr(config-class-map)# set class-default ip-precedence 5
```

### set cos

Данной командой задается значение 802.1p приоритета, которое будет установлено в пакетах, соответствующих конфигурируемому классу.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
set cos <COS>  
no set cos
```

### Параметры

<COS> – классификатор обслуживания в теге 802.1q пакета, принимает значения [0..7].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-CLASS-MAP

CONFIG-CLASS-POLICY-MAP

### Пример

```
esr(config-class-map)# set cos 5
```

```
esr(config-class-policy-map)# set cos 5
```

## set dscp

Данной командой задается значение кода DSCP, которое будет установлено в IP-пакетах, соответствующих конфигурируемому классу.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
set dscp <DSCP>
```

```
no set dscp
```

### Параметры

<DSCP> – значение кода DSCP, принимает значения [0..63].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-CLASS-MAP

CONFIG-CLASS-POLICY-MAP

### Пример

```
esr(config-class-map)# set dscp 16
```

```
esr(config-class-policy-map)# set dscp 5
```

## set ip-precedence

Данной командой задается значение кода IP Precedence, которое будет установлено в IP-пакетах, соответствующих конфигурируемому классу.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
set ip-precedence <IPP>
```

```
no set ip-precedence
```

### Параметры

<IPP> – значение кода IP Precedence, принимает значения [0..7].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-CLASS-MAP


CONFIG-CLASS-POLICY-MAP

## Пример

```
esr(config-class-map)# set ip-precedence 5
```

```
esr(config-class-policy-map)# set ip-precedence 5
```

## set queue

 В текущей версии ПО данный функционал поддерживается только на маршрутизаторах ESR-1000/1200/1500/1700.

Данной командой задается номер выходной очереди QoS, в которую будут направлены пакеты, соответствующие конфигурируемому классу.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
set queue <QUEUE>
```

```
no set queue
```

## Параметры

<QUEUE> – номер выходной аппаратной очереди QoS, принимает значения [1..8].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-CLASS-MAP

## Пример

```
esr(config-class-map)# set queue 5
```

## shape auto-distribution

Данной командой включается автоматическое распределение полосы пропускания между классами, в которых нет настройки полосы пропускания, включая класс по умолчанию.

Использование отрицательной формы команды (no) отключает автоматическое распределение полосы.

### Синтаксис

```
[no] shape auto-distribution
```

### Параметры

Команда не имеет параметров.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-POLICY-MAP

### Пример

```
esr(config-policy-map)# shape auto-distribution
```

## shape average

Данной командой устанавливается ограниченная и гарантированная полоса исходящего трафика для определенного класса или политики в целом.

При использовании команды совместно с командой shape peak для взвешенного класса полоса трафика ограничивается командой shape peak.

Использование отрицательной формы команды (no) снимает ограничение.

### Синтаксис

```
shape average { <BANDWIDTH> | percent <BANDWIDTH_PERCENT> } [<BURST>]
```

```
no shape average
```

### Параметры

<BANDWIDTH> – гарантированная полоса трафика в Кбит/с, принимает значение [1..10000000];

<BURST> – размер сдерживающего порога в байтах, принимает значение:

- для ESR-10/12V/12VF/15/15R/15VF/20/21/30/31/100/200/1000 – [128..16000000];
- для ESR-1200/1500 – [1000..4096000].

<BANDWIDTH\_PERCENT> – гарантированная полоса трафика в %, рассчитывается от (в порядке от более приоритетного к менее приоритетному значению):

- значения shape average корневой политики;
- значения traffic-shape на сетевом интерфейсе, bridge, туннеле;

- значения speed сетевого интерфейса.

Принимает значение [1..100].

### Значение по умолчанию

<BANDWIDTH> – отсутствует;

<BURST> – 128000.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-POLICY-MAP

CONFIG-CLASS-POLICY-MAP

### Пример

```
esr(config-policy-map)# shape average 100000 2000
```

### shape peak

Устанавливается разделяемая полоса исходящего трафика для определенного класса. Данную полосу класс может занять, если другие классы не заняли свою гарантированную полосу. Конфигурация актуальна только для классов работающих во взвешенном режиме (WRR).

Использование отрицательной формы команды (no) снимает ограничение.

### Синтаксис

```
shape peak { <BANDWIDTH> | percent <PERCENT> } [<BURST>]
```

```
no shape peak
```

### Параметры

<BANDWIDTH> – разделяемая полоса трафика в кбит/с, принимает значение [1..10000000];

<BURST> – размер сдерживающего порога в байт, принимает значение:

- для ESR-10/12V/12VF/15/15R/15VF/20/21/30/31/100/200/1000 – [128..16000000];
- для ESR-1200/1500 – [1000..4096000].

<BANDWIDTH\_PERCENT> – общая для priority class полоса трафика в %, конкуренция происходит на основании приоритета класса, рассчитывается от (в порядке от более приоритетного к менее приоритетному значению):

- значения shape average корневой политики;
- значения traffic-shape на сетевом интерфейсе, bridge, туннеле;
- значения speed сетевого интерфейса.

Принимает значение [1..100].



### Значение по умолчанию

<BANDWIDTH> – отсутствует;

<BURST> – 128000.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-CLASS-POLICY-MAP

### Пример

```
esr(config-class-policy-map)# shape average 100000 2000
```

### show qos interface shapers

Данная команда показывает параметры QoS сетевых интерфейсов.

### Синтаксис

```
show qos interface shapers <IF>
```

### Параметры

<IF> – интерфейс или группы интерфейсов, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
esr# show qos interface shapers gigabitethernet 1/0/2
gigabitethernet 1/0/2
Committed rate: 100000 Kbps
Committed burst: 1600 KBytes
```

### show qos map cos-queue

Данная команда показывает информацию о соответствии кодов 802.1p в пакетах и выходных очередях, используемых в QoS.

**Синтаксис**

```
show qos map cos-queue
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример**

```
esr# show qos map dscp-queue
d1 : d2   0  1  2  3  4  5  6  7
-----
0         01 02 03 04 05 06 07 08
```

**show qos map dscp-mutation**

Данная команда показывает информацию о соответствии кодов DSCP в пакетах и кодов DSCP после изменений.

**Синтаксис**

```
show qos map dscp-mutation
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример**

```
esr# show qos map dscp-mutation
d1 : d2   0  1  2  3  4  5  6  7  8  9
-----
0         00 01 02 03 04 05 06 07 08 09
1         10 11 12 13 14 15 16 17 18 19
2         20 21 22 23 24 25 26 27 28 29
3         30 31 32 33 34 35 36 37 38 39
```

```

4          40 41 42 43 44 45 46 47 48 49
5          50 51 52 53 54 55 56 57 58 59
6          60 61 62 63

```

## show qos map dscp-queue

Данная команда показывает информацию о соответствии кодов DSCP в пакетах и выходных очередях, используемых в QoS.

### Синтаксис

```
show qos map dscp-queue
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```

esr# show qos map dscp-queue
d1 : d2  0  1  2  3  4  5  6  7  8  9
-----  -
0         01 01 01 01 01 01 01 01 02 02
1         02 02 02 02 02 02 03 03 03 03
2         03 03 03 03 04 04 04 04 04 04
3         04 04 05 05 05 05 05 05 05 05
4         06 06 06 06 06 06 06 06 07 07
5         07 07 07 07 07 07 08 08 08 08
6         08 08 08 08

```

## show qos policy binding

Данная команда используется для просмотра привязанной QoS-политики к заданному интерфейсу для классификации входящего (input) или приоритизации исходящего (output) трафика.

### Синтаксис

```
show qos policy binding [ <IF> | <TUN> ]
```

### Параметры

<IF> – интерфейс или группы интерфейсов, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<TUN> – имя туннеля, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#).

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример**

```

esr# show qos policy binding gi1/0/1
gigabitethernet 1/0/1
Output: parent
  Class: class1
    Policy: child1
      Class: class3
  Class: class2
    Policy: child2

```

**show qos policy configuration**

Данная команда выводит конфигурацию заданной политики QoS.

**Синтаксис**

```
show qos policy configuration <NAME>
```

**Параметры**

<NAME> – имя политики, задается строкой до 31 символа.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример**

```

esr# show qos policy configuration parent
Policy-map: parent
  Shape average:
    Bandwidth:      --
    Burst:          128
  Class: class1
    Set:
      COS:          --
      DSCP:         --
      IP-Precedence: --
      Queue:        --
    Match access group: --
    Shape average:

```

```

    Bandwidth:      --
    Burst:          128
  Shape peak:
    Bandwidth:      --
    Burst:          128
  Mode:            FIFO
  Priority class:   8
  Queue limit:     127
  Service policy:  child1
Class: class2
  Set:
    COS:            --
    DSCP:           --
    IP-Precedence:  --
    Queue:          --
  Match access group: --
  Shape average:
    Bandwidth:      --
    Burst:          128
  Shape peak:
    Bandwidth:      --
    Burst:          128
  Mode:            FIFO
  Priority class:   8
  Queue limit:     127
  Service policy:  child2

```

## show qos policy statistics

Данная команда выводит статистику по переданным и отброшенным пакетам согласно политике, назначенной на output. Команда актуальна только для Policy-based QoS режима интерфейса.

### Синтаксис

```
show qos policy statistics [ <IF> | <TUN> ]
```

### Параметры

<IF> – интерфейс или группы интерфейсов, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<TUN> – имя туннеля, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#).

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```

esr# sh qos policy statistics
gigabitethernet 1/0/2
  Policy пом-1

```

```

Input policy root
Input class root
Shape: bytes 750947679, packets 496667, drops 1002200
  Class clm-1
    Shape: bytes 750946896, packets 496658, drops 1002200
  Class class-default
    Shape: bytes 783, packets 9, drops 0

```

## show qos statistics

Данная команда выводит статистику по переданным и отброшенным пакетам. Команда актуальна только для basic QoS режима интерфейса.

### Синтаксис

```
show qos statistics [ <IF> | <TUN> ]
```

### Параметры

<IF> – интерфейс или группы интерфейсов, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<TUN> – имя туннеля, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#).

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```

esr# show qos statistics vti 2
vti 2
Queue   Bytes           Packets           Drops
-----
1        0                0                 0
2        0                0                 0
3        0                0                 0
4        0                0                 0
5        0                0                 0
6        964073836       1413598           0
7        121389180       177990            1235497
8        0                0                 0

```

## show qos tunnel shapers

Данная команда показывает параметры QoS-туннелей.

### Синтаксис

```
show qos tunnel shapers <TUN>
```

## Параметры

<TUN> – имя туннеля, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#).

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
esr# show qos tunnel shapers vti 2
vti 2
qid   Target          Target
      Committed      Committed
      Rate [Kbps]    Burst [KBytes]
----  -
1     10000            128
2     6000             128
```

## traffic-shape

Данная команда устанавливает ограничение скорости исходящего трафика для определенной очереди или интерфейса в целом. Команда актуальна только для Basic QoS режима интерфейса.

Использование отрицательной формы команды (no) снимает ограничение.

## Синтаксис

```
traffic-shape { <BANDWIDTH> [BURST] | queue <QUEUE> <BANDWIDTH> [BURST] }
no traffic-shape [ queue <QUEUE> ]
```

## Параметры

<QUEUE> – идентификатор очереди, принимает значение [1..8];

<BANDWIDTH> – средняя скорость трафика в кбит/с, принимает значения:

- для интерфейсов gigabitethernet, loopback, e1: [1..10000000];
- для интерфейсов tengigabitethernet: [3000..100000000];
- для интерфейсов fortygigabitethernet: [3000..40000000];
- для интерфейсов loopback:

<BURST> – размер сдерживающего порога в кБ, принимает значение [4..16000]. Возможно указывать значение только кратное 4.

## Значение по умолчанию

<BANDWIDTH> – отсутствует;

<BURST> – 128.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-CELLULAR-MODEM

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-IF-E1

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-VTI

CONFIG-L2TPV3

CONFIG-PPPOE

CONFIG-PPTP

CONFIG-L2TP

CONFIG-IF-MULTILINK

CONFIG-OPENVPN

**Пример**

```
esr(config-if)# traffic-shape queue 3 100000 2000
```



## 28 Мониторинг и управление

- Управление Netflow
- Управление sFlow
- Управление SNMP
- Управление SYSLOG
- Настройка доступа SSH, Telnet
- Настройка протокола LLDP
- Настройка зеркалирования
- Настройка zabbix-agent и zabbix-proxy

### Управление Netflow

- `ip netflow export`
- `netflow active-timeout`
- `netflow collector`
- `netflow domain-id`
- `netflow enable`
- `netflow export-events web-proxy`
- `netflow export-options`
- `netflow inactive-timeout`
- `netflow max-flows`
- `netflow refresh-rate`
- `netflow version`
- `netflow self-ifindex`
- `port`
- `show netflow configuration`
- `show netflow statistics`
- `show netflow statistics cpu`
- `source-address`

#### `ip netflow export`

Данная команда используется для включения экспорта статистики Netflow на сетевом интерфейсе.

Использование отрицательной формы команды (`no`) отключает экспорт статистики Netflow на сетевом интерфейсе.

#### Синтаксис

```
ip netflow export [ <DIRECTION> ]
no ip netflow export
```

#### Параметры

<DIRECTION> – направление трафика, по которому собирается статистика, принимает значения:

- `ingress` – статистика собирается только для входящего трафика;
- `egress` – статистика собирается только для исходящего трафика;
- `both` – статистика собирается для входящего и исходящего трафика.

#### Значение по умолчанию

`both`

## Необходимый уровень привилегий

15

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-CELLULAR-MODEM

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-VTI

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-BRIDGE

CONFIG-LT

CONFIG-PPPOE

### Пример

```
esr(config-if-gi)# ip netflow export
```

### netflow active-timeout

Данной командой задаётся интервал, по истечении которого информация об активных сессиях экспортируется на коллектор.

В случае если значение параметра active-timeout превышает значение параметра inactive-timeout, то вместо экспортирования будет происходить накопление данных с заданной периодичностью active-timeout.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
netflow active-timeout <TIMEOUT>
```

```
no netflow active-timeout
```

## Параметры

<TIMEOUT> – задержка перед отправкой информации об активных сессиях, задается в секундах, принимает значение [5..36000].

## Значение по умолчанию

1800 секунд.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# netflow active-timeout 30
```

## netflow collector

Данная команда используется для создания коллектора Netflow и перехода в командный режим CONFIG-NETFLOW-HOST.

Использование отрицательной формы команды (no) удаляет сконфигурированный коллектор Netflow.

## Синтаксис

```
[no] netflow collector <ADDR> [ vrf <VRF> ]
```

## Параметры

<ADDR> – IP-адрес коллектора, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# netflow collector 10.100.100.1  
esr(config-netflow-host)#
```

**netflow domain-id**

Данной командой устанавливается значение Observation Domain ID в пакетах Netflow.  
Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
netflow domain-id <ID>  
no netflow domain-id
```

**Параметры**

<ID> – значение Observation Domain ID, принимает значение [0.. 4294967295].

**Значение по умолчанию**

0

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# netflow domain-id 35
```

**netflow enable**

Данной командой активируется Netflow на маршрутизаторе.  
Использование отрицательной формы команды (no) деактивирует Netflow на маршрутизаторе.

**Синтаксис**

```
[no] netflow enable
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Процесс выключен.

**Необходимый уровень привилегий**

15

## Командный режим

CONFIG

### Пример

```
esr(config)# netflow enable
```

### **netflow export-events web-proxy**

Данная команда позволяет включить передачу netflow-статистики о пакетах, переданных через web-proxy.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
[no] netflow export-events web-proxy
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

15

## Командный режим

CONFIG

### Пример

```
esr(config)# netflow export-events web-proxy
```

### **netflow export-options**

Данная команда позволяет включить дополнительные поля в экспортируемую на коллектор информацию.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
[no] netflow export-options { http-host | http-url }
```

## Параметры

`http-host` – для HTTP/HTTPS-сессий будет добавляться информация о домене хоста, на который был отправлен HTTP-запрос. Поле в Netflow-пакете имеет идентификатор 24884, размерность 128 байт.

`http-url` – для HTTP-сессий будет добавляться информация о URL, на который был отправлен HTTP-запрос. Поле в Netflow-пакете имеет идентификатор 24887, размерность 384 байта.

## Значение по умолчанию

Отключено.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# netflow export-options http-host
```

## `netflow inactive-timeout`

Данной командой задаётся интервал, по истечении которого информация об устаревших сессиях экспортируется на коллектор.

Использование отрицательной формы команды (`no`) устанавливает значение по умолчанию.

## Синтаксис

```
netflow inactive-timeout <TIMEOUT>
```

```
no netflow inactive-timeout
```

## Параметры

`<TIMEOUT>` – задержка перед отправкой информации об устаревших сессиях, задается в секундах, принимает значение [0..240].

## Значение по умолчанию

15

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# netflow inactive-timeout 30
```

## netflow max-flows

Данной командой задаётся максимальное количество наблюдаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
netflow max-flows <COUNT>
```

```
no netflow max-flows
```

## Параметры

<COUNT> – количество наблюдаемых сессий, принимает значение [10000..2000000].

## Значение по умолчанию

512000

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# netflow max-flows 300000
```

## netflow refresh-rate

Данной командой задаётся частота отправки шаблонов на Netflow-коллектор.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
netflow refresh-rate <RATE>
```

```
no netflow refresh-rate
```

## Параметры

<RATE> – частота отправки статистики, задается в пакетах на поток, принимает значение [1..10000].

**Значение по умолчанию**

10

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# netflow refresh-rate 100
```

**netflow version**

Данной командой задаётся версия Netflow-протокола.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
netflow version <VERSION>
```

```
no netflow version
```

**Параметры**

<VERSION> – версия Netflow-протокола: 5, 9 и 10.

**Значение по умолчанию**

9

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# netflow version 10
```

**netflow self-ifindex**

Данной командой устанавливается ifindex для трафика, принадлежащего маршрутизатору, или для трафика, который был отброшен в результате обработки маршрутизатором. В результате выполнения



команды в статистике, отправленной на коллектор, ifindex для данного вида трафика будет соответствовать настроенному.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
netflow self-ifindex <INDEX>  
no netflow version
```

### Параметры

<INDEX> – 4294967295 или 0.

### Значение по умолчанию

4294967295

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

### Пример

```
esr(config)# netflow self-ifindex 0
```

### port

Данной командой определяется порт Netflow-сервиса на сервере сбора статистики.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
port <PORT>  
no port
```

### Параметры

<PORT> – номер UDP-порта, указывается в диапазоне [1..65535].

### Значение по умолчанию

2055

### Необходимый уровень привилегий

15

## Командный режим

CONFIG-NETFLOW-HOST

### Пример

```
esr(config-netflow-host)# port 5555
```

## show netflow configuration

Командой выполняется просмотр параметров конфигурации Netflow-агента.

### Синтаксис

```
show netflow configuration
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

1

## Командный режим

ROOT

### Пример

```
esr# show netflow configuration
Netflow configuration:
Global state:      Enabled
Version:           9
Domian ID:         0
Maxflows:          512000
Refresh rate:      10
Self ifindex:      4294967295
Active timeout:    1800
Inactive timeout:  15
Export-options:

Collectors:
  192.168.39.1:2055
```

## show netflow statistics

Команда для просмотра текущей информации о работе Netflow.

### Синтаксис

```
show netflow statistics
```

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```

esr# show netflow statistics
NAT events: Enabled; count start 0; count stop 0
Flows:      Active 0; peak (0 reached 55 seconds ago); memory 3931 Kbytes
Hash:      Size 503185 Kbytes; memory 3931 Kbytes
InHash:    Packets 0; size 0 Kbytes
InPDU:    Packets 0; size 0 bytes

Processed rate  Bits/s      Packets/s
-----
Current        0             0
1 Min Avg      0             0
5 Min Avg      0             0

```

## show netflow statistics cpu

Команда для просмотра статистики по распределению информации о нагрузке Netflow на ЦПУ.

## Синтаксис

```
show netflow statistics
```

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```

esr# show netflow statistics cpu
Export:
Rate:      0 bytes/s
Total:    Packets 2; size 0 Mbytes; flows 0

```

```
Errors:      Packets 0
Traffic lost: Packets 0; size 0 Kbytes; flows 0
```

CPU	Rate, Packets/s	Traffic, Packets	Traffic, MBytes	Drop, Packets	Drop, KBytes
0	0	0	0	0	0
1	0	0	0	0	0
2	0	0	0	0	0
3	0	0	0	0	0
Summary	0	0	0	0	0

### source-address

Данной командой определяется IP-адрес маршрутизатора, который будет использоваться в качестве IP-адреса источника в отправляемых Netflow пакетах.

Использование отрицательной формы команды (no) удаляет указанный IP-адрес источника.

### Синтаксис

```
source-address <ADDR>
no source-address
```

### Параметры

<ADDR> – IP-адрес источника, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-NETFLOW-HOST

### Пример

```
esr(config-netflow-host)# source-address 10.100.100.2
```

## Управление sFlow

- [ip sflow export](#)
- [port](#)
- [sflow agent-ip](#)
- [sflow collector](#)
- [sflow enable](#)
- [sflow poll-interval](#)
- [sflow sampling-rate](#)
- [show sflow configuration](#)

### ip sflow export

Данная команда используется для включения экспорта статистики sFlow на сетевом интерфейсе. Функция sFlow на сетевом интерфейсе может быть включена, если на интерфейсе выключена функция Firewall (раздел [ip firewall disable](#)), в ином случае экспорт статистики sFlow настраивается в правиле Firewall (раздел [action](#)).

Использование отрицательной формы команды (no) отключает экспорт статистики sFlow на сетевом интерфейсе.

### Синтаксис

```
[no] ip sflow export
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-CELLULAR-MODEM

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-VTI

CONFIG-GRE

CONFIG-IP4IP4  
CONFIG-BRIDGE  
CONFIG-LT  
CONFIG-PPPOE

### Пример

```
esr(config-if-gi)# ip sflow export
```

### port

Данной командой определяется порт sFlow-сервиса на сервере сбора статистики.  
Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
port <PORT>  
no port
```

### Параметры

<PORT> – номер UDP-порта, указывается в диапазоне [1..65535].

### Значение по умолчанию

6343

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-SFLOW-HOST

### Пример

```
esr(config-sflow-host)# port 5556
```

### sflow agent-ip

Данная команда используется для назначения адреса sFlow agent. Если команда не указана, то в качестве адреса sFlow agent будет использован случайный адрес из присутствующих в конфигурации.  
Использование отрицательной формы команды (no) удаляет настройку адреса sFlow agent.

### Синтаксис

```
sflow agent-ip <ADDR>  
[no] sflow agent-ip
```

## Параметры

<ADDR> – IPv4/IPv6-адрес. При использовании IPv4-адресации задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Для IPv6 задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# sflow agent-ip 192.168.44.18
```

## sflow collector

Данная команда используется для создания коллектора sFlow и перехода в командный режим CONFIG-SFLOW-HOST.

Использование отрицательной формы команды (no) удаляет сконфигурированный коллектор sFlow.

## Синтаксис

```
[no] sflow collector <ADDR> [ vrf <VRF> ]
```

## Параметры

<ADDR> – IP-адрес коллектора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# sflow collector 10.100.100.1  
esr(config-sflow-host)#
```

## **sflow enable**

Данной командой активируется sFlow на маршрутизаторе.

Использование отрицательной формы команды (no) деактивирует sFlow на маршрутизаторе.

### **Синтаксис**

```
[no] sflow enable
```

### **Параметры**

Команда не содержит параметров.

### **Значение по умолчанию**

Процесс выключен.

### **Необходимый уровень привилегий**

15

### **Командный режим**

CONFIG

### **Пример**

```
esr(config)# sflow enable
```

## **sflow poll-interval**

Данной командой задаётся интервал, по истечении которого происходит получение информации о счетчиках сетевого интерфейса.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### **Синтаксис**

```
sflow poll-interval <TIMEOUT>
```

```
no sflow poll-interval
```

### **Параметры**

<TIMEOUT> – интервал, по истечении которого происходит получение информации о счетчиках сетевого интерфейса, принимает значение [1..300].

### **Значение по умолчанию**

10 секунд



**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# sflow poll-interval 30
```

**sflow sampling-rate**

Данной командой задаётся частота отправки пакетов пользовательского трафика в неизменном виде на sFlow-коллектор.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
sflow sampling-rate <RATE>
```

```
no sflow sampling-rate
```

**Параметры**

<RATE> – частота отправки пакетов пользовательского трафика на коллектор, принимает значение [1..65535]. При значении частоты 10 на коллектор будет отправлен один пакет из десяти.

**Значение по умолчанию**

1000

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# sflow sampling-rate 100
```

**show sflow configuration**

Командой выполняется просмотр параметров конфигурации sFlow-агента.

## Синтаксис

```
show sflow configuration
```

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
esr# show sflow configuration
sFlow configuration:
Global state:   Enabled
Sample rate:   1000
Poll interval: 10
Agent:         192.168.44.18

Collectors:
  10.100.100.1:5556
```

## Управление SNMP

- [access](#)
- [authentication access](#)
- [authentication algorithm](#)
- [authentication key](#)
- [client-list](#)
- [community](#)
- [ip address](#)
- [ipv6 address](#)
- [enable](#)
- [oid-tree](#)
- [port](#)
- [privacy algorithm](#)
- [privacy key](#)
- [rmon collection statistics](#)
- [snmp-server](#)
- [snmp-server community](#)
- [snmp-server contact](#)
- [snmp-server dscp](#)
- [snmp-server enable traps](#)
- [snmp-server enable traps config](#)
- [snmp-server enable traps entity](#)
- [snmp-server enable traps entity-sensor](#)
- [snmp-server enable traps environment](#)
- [snmp-server enable traps envmon](#)
- [snmp-server enable traps files-operations](#)
- [snmp-server enable traps flash](#)
- [snmp-server enable traps interfaces](#)
- [snmp-server enable traps ports](#)
- [snmp-server enable traps screens](#)
- [snmp-server enable traps snmp](#)
- [snmp-server enable traps syslog](#)
- [snmp-server enable traps wifi](#)
- [snmp-server host](#)
- [snmp-server location](#)
- [snmp-server system-shutdown](#)
- [snmp-server trap link](#)
- [snmp-server user](#)
- [snmp-server view](#)
- [source-address](#)
- [source-interface](#)
- [view](#)

### access

Данной командой определяется уровень доступа по протоколу SNMPv3.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
access <TYPE>
```

```
no access
```

## Параметры

<TYPE> – уровень доступа:

- ro – доступ только для чтения;
- rw – доступ для чтения и записи.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-SNMP-USER

## Пример

```
esr(config-snmp-user)# access rw
```

## authentication access

Данной командой определяется режим безопасности.

Использование отрицательной формы команды (no) отключает аутентификацию.

## Синтаксис

```
authentication access <TYPE>
```

```
no authentication access
```

## Параметры

<TYPE> – режим безопасности:

- auth – используется только аутентификация;
- priv – используется аутентификация и шифрование данных.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-SNMP-USER

## Пример

```
esr(config-snmp-user)# authentication algorithm auth
```

## authentication algorithm

Данная команда определяет алгоритм аутентификации SNMPv3-запросов.

Использование отрицательной формы команды (no) отключает аутентификацию.

### Синтаксис

```
authentication algorithm <ALGORITHM>
```

```
no authentication algorithm
```

### Параметры

<ALGORITHM> – алгоритм шифрования:

- md5 – пароль шифруется по алгоритму md5;
- sha1 – пароль шифруется по алгоритму sha1.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-SNMP-USER

### Пример

```
esr(config-snmp-user)# authentication algorithm md5
```

### authentication key

Данная команда устанавливает пароль для аутентификации SNMPv3-запросов.

Использование отрицательной формы команды (no) удаляет пароль.

### Синтаксис

```
authentication key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
```

```
no authentication key
```

### Параметры

<CLEAR-TEXT> – пароль, задаётся строкой от 8 до 32 символов;

encrypted – при указании команды задаётся зашифрованный пароль:

<ENCRYPTED-TEXT> – зашифрованный пароль размером от 8 байт до 32 байт (от 16 до 64 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-SNMP-USER

## Пример

```
esr(config-snmp-user)# authentication key ascii-text 123456789
esr(config-snmp-user)# authentication key ascii-text encrypted CDE65039E5591FA3F1
```

## client-list

Данной командой активируется фильтрация и устанавливается профиль IP-адресов, с которых могут приниматься SNMPv3-пакеты с данным именем SNMPv3-пользователя.

Использование отрицательной формы команды (no) деактивирует фильтрацию принимаемых SNMPv3-пакетов.

## Синтаксис

```
[no] client-list <NAME>
```

## Параметры

<NAME> – имя ранее созданной object-group, задается строкой до 31 символа.

## Значение по умолчанию

Ограничения отключены.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-SNMP-USER

## Пример

```
esr(config-snmp-user)# client-list OBG005
```

## community

Данной командой определяется SNMP-community для отправки уведомлений на удаленный сервер.

Использование отрицательной формы команды (no) удаляет значение community.

## Синтаксис

```
community <COMMUNITY>
```

```
no community
```

## Параметры

<COMMUNITY> – сообщество для доступа по протоколу SNMP, задается строкой длиной [1..128] символа.

**Значение по умолчанию**

community – не задано.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-SNMP-HOST

**Пример**

```
esr(config-snmp-host)# community privatekey
```

**ip address**

Данной командой активируется фильтрация и устанавливается IP-адрес, которому предоставлен доступ к маршрутизатору под данным SNMPv3-пользователем.

Использование отрицательной формы команды (no) деактивирует фильтрацию принимаемых SNMPv3-пакетов.

**Синтаксис**

```
[no] ip address <ADDR>
```

**Параметры**

<ADDR> – IP-адрес клиента, которому предоставлен доступ, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

**Значение по умолчанию**

Ограничения отключены.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-SNMP-USER

**Пример**

```
esr(config-snmp-user)# ip address 192.168.85.33
```

## ipv6 address

Данной командой активируется фильтрация и устанавливается IPv6-адрес, которому предоставлен доступ к маршрутизатору под данным SNMPv3-пользователем.

Использование отрицательной формы команды (no) деактивирует фильтрацию принимаемых SNMPv3-пакетов.

### Синтаксис

```
[no] ipv6 address <IPV6-ADDR>
```

### Параметры

<IPV6-ADDR> – IPv6-адрес клиента, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-SNMP-USER

### Пример

```
esr(config-snmp-user)# ipv6 address AC:05:12:44::24
```

## enable

Данной командой активируется SNMPv3-пользователь.

Использование отрицательной формы команды (no) деактивирует SNMPv3-пользователя.

### Синтаксис

```
[no] enable
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

15



## Командный режим

CONFIG-SNMP-USER

### Пример

```
esr(config-snmp-user)# enable
```

### oid-tree

Данной командой устанавливается OID и действие, применяемое к нему (разрешить/запретить). Более длинные OID имеют преимущество.

OID указывается в цифровой нотации.

Использование отрицательной формы команды (no) удаляет запись oid-tree.

### Синтаксис

```
oid-tree <OID> <ACTION>
```

```
no oid-tree <OID>
```

### Параметры

<OID> – OID, задаётся строкой до 255 символов;

<ACTION> – действие, применяемое к OID:

- excluded – запретить использование OID;
- included – разрешить использование OID.

### Значение по умолчанию

Разрешено.

### Необходимый уровень привилегий

15

## Командный режим

CONFIG-SNMP-VIEW

### Пример

```
esr(config-snmp-view)# oid-tree 1.3.6.1.2.1.2.2 excluded
```

### port

Данной командой определяется порт коллектора SNMP уведомлений на удаленном сервере.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
port <PORT>  
no port
```

**Параметры**

<PORT> – номер UDP-порта, указывается в диапазоне [1..65535].

**Значение по умолчанию**

162

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-SNMP-HOST

**Пример**

```
esr(config-snmp-host)# port 5555
```

**privacy algorithm**

Данная команда определяет алгоритм шифрования передаваемых данных.  
Использование отрицательной формы команды (no) отключает шифрование.

**Синтаксис**

```
privacy algorithm <ALGORITHM>  
no privacy algorithm
```

**Параметры**

<ALGORITHM> – алгоритм шифрования:

- aes128 – использовать алгоритм шифрования AES-128;
- des – использовать алгоритм шифрования DES.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-SNMP-USER

## Пример

```
esr(config-snmp-user)# privacy algorithm des
```

## privacy key

Данная команда устанавливает пароль для шифрования передаваемых данных.

Использование отрицательной формы команды (no) удаляет пароль.

## Синтаксис

```
privacy key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no privacy key
```

## Параметры

<CLEAR-TEXT> – пароль, задаётся строкой от 8 до 32 символов;

<ENCRYPTED-TEXT> – зашифрованный пароль размером от 8 байт до 32 байт (от 16 до 64 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-SNMP-USER

## Пример

```
esr(config-snmp-user)# privacy key ascii-text 123456789
esr(config-snmp-user)# privacy key ascii-text encrypted CDE65039E5591FA3F1
```

## rmon collection statistics

Данная команда включает сохранение RMON-статистики для физического интерфейса.

Использование отрицательной формы команды (no) отключает сохранение RMON-статистики для физического интерфейса.

## Синтаксис

```
rmon collection statistics <INDEX> owner <OWNER>
no rmon collection statistics
```

## Параметры

<INDEX> – RMON-индекс данного интерфейса;

<OWNER> – текстовое поле длиной [1..127] символов, описывающее владельца, создавшего данный процесс.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

**Пример**

```
esr(config-if)# rmon collection statistics 17 owner admin
```

**snmp-server**

Данной командой включается SNMP-сервер как в глобальной таблице маршрутизации, так и во всех созданных VRF.

Использование отрицательной формы команды (no) выключает SNMP-сервер.

**Синтаксис**

```
[no] snmp-server
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Выключен.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# snmp-server
```

**snmp-server community**

Данной командой определяется сообщество для доступа по протоколу SNMP.

Использование отрицательной формы команды (no) удаляет настройки сообщества.

## Синтаксис

```
[no] snmp-server community <COMMUNITY> [ <TYPE> ] [ { <ADDR> | <IPV6-ADDR> } ] [client-list <OBJ-GROUP-NETWORK-NAME> ] [ <VERSION> ] [ view <VIEW-NAME> ] [ vrf <VRF> ]
```

## Параметры

<COMMUNITY> – сообщество для доступа по протоколу SNMP, задается строкой длиной [1..128] символа;

<TYPE> – уровень доступа:

- ro – доступ только для чтения;
- rw – доступ для чтения и записи.

<ADDR> – IP-адрес клиента, которому предоставлен доступ, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес клиента, задается в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, от которых обрабатываются snmp-запросы, задается строкой до 31 символа;

<VERSION> – версия snmp, поддерживаемая данным community, принимает значения v1 или v2c;

<VIEW-NAME> – имя SNMP view профиля, на основании которого обеспечивается доступ к OID;

<VRF> – имя экземпляра VRF, из которого будет разрешен доступ, задается строкой до 31 символа.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# snmp-server community public rw
```

## snmp-server contact

Данной командой устанавливается значение переменной SNMP, содержащей контактную информацию (по умолчанию не определена). Для удобства в параметрах можно указать ответственного за данное оборудование, например, его фамилию.

Использование отрицательной формы команды (no) удаляет значение переменной SNMP, содержащей контактную информацию.

## Синтаксис

```
[no] snmp-server contact <CONTACT>
```

## Параметры

<CONTACT> – контактная информация, задаётся строкой до 255 символов.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# snmp-server contact ivanov_ivan
```

## snmp-server dscp

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов SNMP-сервера.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

## Синтаксис

```
snmp-server dscp <DSCP>
```

```
no snmp-server dscp
```

## Параметры

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

## Значение по умолчанию

61

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# snmp-server dscp 40
```

## snmp-server enable traps

Данная команда разрешает отправку всех типов SNMP-уведомлений.

Использование отрицательной формы команды (no) отменяет разрешение отправки всех типов SNMP-уведомлений.

### Синтаксис

```
[no] snmp-server enable traps
```

### Параметры

Отсутствуют.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# snmp-server enable traps
```

### snmp-server enable traps config

Данная команда разрешает отставку SNMP-уведомлений об операциях с конфигурацией.

Использование отрицательной формы команды (no) отменяет разрешение отправки указанных типов уведомлений.

### Синтаксис

```
[no] snmp-server enable traps config [ <ACT> ]
```

### Параметры

<ACT> – трапы фактов изменения конфигурации:

- commit – применение изменения конфигурации;
- confirm – подтверждение изменения конфигурации.

Без указания ключа <ACT> – активируется отправка всех трапов данной группы.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# snmp-server enable traps config commit
```

### **snmp-server enable traps entity**

Данная команда разрешает отправку SNMP-уведомлений об операциях с running-config.

Использование отрицательной формы команды (no) отменяет разрешение отправки указанных типов уведомлений.

#### **Синтаксис**

```
[no] snmp-server enable traps entity [ <ENT> ]
```

#### **Параметры**

<ENT> – типы фильтров параметров окружения:

- config-change – информация о операциях с running-config.

Без указания ключа <ENT> – активируется отправка всех трапов данной группы.

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG

#### **Пример**

```
esr(config)# snmp-server enable traps entity
```

### **snmp-server enable traps entity-sensor**

Данная команда разрешает отправку SNMP-уведомлений об изменении параметров окружения.

Использование отрицательной формы команды (no) отменяет разрешение отправки указанных типов уведомлений.

#### **Синтаксис**

```
[no] snmp-server enable traps entity-sensor [ <ENT> ]
```

#### **Параметры**

<ENT> – типы фильтров параметров окружения:

- threshold – информация о срабатывании пересечения пороговых значений.

Без указания ключа <ENT> – активируется отправка всех трапов данной группы.

#### **Необходимый уровень привилегий**

10



## Командный режим

CONFIG

### Пример

```
esr(config)# snmp-server enable traps entity-sensor
```

### snmp-server enable traps environment

Данная команда разрешает отправку SNMP-уведомлений об изменении параметров окружения.

Использование отрицательной формы команды (no) отменяет разрешение отправки указанных типов уведомлений.

### Синтаксис

```
[no] snmp-server enable traps environment [ <ENV> ]
```

### Параметры

<ENV> – типы фильтров параметров окружения:

- pwrin – отказ БП;
- pwrin-insert – БП установлен;
- fan – отказ вентилятора;
- fan-speed-changed – изменение скорости вентиляторов;
- fan-speed-high – скорость вращения вентиляторов превысила максимальный порог;
- memory-flash-low – свободный объем NAND меньше заданного порога;
- memory-flash-critical-low – свободный объем NAND меньше заданного критического порога;
- memory-ram-low – свободный объем RAM меньше заданного порога;
- memory-ram-critical-low – свободный объем RAM меньше заданного критического порога;
- cpu-load – высокая нагрузка ЦПУ;
- cpu-overheat-temp – температура CPU превысила заданный максимальный порог;
- cpu-critical-temp – температура CPU превысила заданный критический порог;
- cpu-supercooling-temp – температура CPU упала ниже заданного минимального порога;
- switch-overheat-temp – температура коммутатора превысила заданный максимальный порог;
- switch-supercooling-temp – температура коммутатора упала ниже заданного минимального порога;
- board-overheat-temp – перегрев платы;
- board-supercooling-temp – переохлаждение платы;
- sfp-overheat-temp – перегрев sfp-модуля;
- sfp-supercooling-temp – переохлаждение sfp-модуля.

Без указания ключа <ENV> – активируется отправка всех трапов данной группы.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# snmp-server enable traps enviroment pwrin
```

### snmp-server enable traps envmon

Данная команда разрешает отправку SNMP-уведомлений об изменении параметров окружения.

Использование отрицательной формы команды (no) отменяет разрешение отправки указанных типов уведомлений.

### Синтаксис

```
[no] snmp-server enable traps envmon [ <ENV> ]
```

### Параметры

<ENV> – типы фильтров параметров окружения:

- fan – информация о работе блоков вентиляторов;
- shutdown – информация о отключении маршрутизатора;
- supply – информация о работе блоков питания;
- temperature информация о работе температурных датчиков.

Без указания ключа <ENV> – активируется отправка всех трапов данной группы.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

## Пример

```
esr(config)# snmp-server enable traps envmon fun
```

### snmp-server enable traps files-operations

Данная команда разрешает отправку SNMP-уведомлений об операциях с файлами.

Использование отрицательной формы команды (no) отменяет разрешение отправки указанных типов уведомлений.

### Синтаксис

```
[no] snmp-server enable traps files-operations [ <ACT> ]
```

## Параметры

<ACT> – типы фильтров параметров операций с файлами:

- successful – успешно;
- failed – неудачно;
- canceled – отменено.

Без указания ключа <ACT> – активируется отправка всех трапов данной группы.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# snmp-server enable traps files-operations canceled
```

## snmp-server enable traps flash

Данная команда разрешает отправку SNMP-уведомлений об операциях с внешними flash-накопителями.

Использование отрицательной формы команды (no) отменяет разрешение отправки указанных типов уведомлений.

## Синтаксис

```
[no] snmp-server enable traps flash [ <ACT> ]
```

## Параметры

<ACT> – типы фильтров параметров операций с файлами:

- insertion – подключение flash-накопителя;
- removal – удаление flash-накопителя.

Без указания ключа <ACT> – активируется отправка всех трапов данной группы.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# snmp-server enable traps flash removal
```

**snmp-server enable traps interfaces**

Данная команда разрешает отправку SNMP-уведомлений о изменении состояния интерфейсов.

Использование отрицательной формы команды (no) отменяет разрешение отправки указанных типов уведомлений.

**Синтаксис**

```
[no] snmp-server enable traps interfaces [ <ACT> ]
```

**Параметры**

<ACT> – типы фильтров параметров окружения:

- rx-utilization-high – поток входящих данных превышает порог;
- tx-utilization-high – поток исходящих данных превышает порог;
- number-high – превышение количества IP-интерфейсов;

Без указания ключа <ACT> – активируется отправка всех трапов данной группы.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# snmp-server enable traps interfaces rx-utilization-high
```

**snmp-server enable traps ports**

Данная команда разрешает отправку SNMP-уведомлений о появлении ошибок на интерфейсах чипа коммутации.

Использование отрицательной формы команды (no) отменяет разрешение отправки указанных типов уведомлений.

**Синтаксис**

```
[no] snmp-server enable traps ports [ <TYPE> ]
```

**Параметры**

<TYPE> – типы фильтров состояние порта:

- port-counters-errors – ошибки на интерфейсах чипа коммутации.

Без указания ключа <TYPE> – активируется отправка всех трапов данной группы.

**Необходимый уровень привилегий**

10

## Командный режим

### CONFIG

#### Пример

```
esr(config)# snmp-server enable traps ports
```

#### snmp-server enable traps screens

Данная команда разрешает отправку SNMP-уведомлений о срабатывании защиты от определенного вида DoS-атак.

Использование отрицательной формы команды (no) отменяет разрешение отправки указанных типов уведомлений.

#### Синтаксис

```
[no] snmp-server enable traps screens [ <SCREEN> ]
```

#### Параметры

<SCREEN> – типы фильтров защиты от DoS-атак:

- dest-limit – ограничение количества одновременных сессий на основании адреса назначения;
- source-limit – ограничение количества одновременных сессий на основании адреса источника;
- icmp-threshold – защита от ICMP flood атак;
- udp-threshold – защита от UDP flood атак;
- syn-flood – защита от SYN flood атак;
- land – защита от land атак;
- winnuke – защита от winnuke атак;
- icmp-frag – блокировка фрагментированных ICMP-пакетов;
- syn-flag – блокировка фрагментированных TCP-пакетов, с флагом SYN;
- unknown-proto – блокировка пакетов, с ID протокола в заголовке IP равном 137 и более;
- ip-frag – блокировка фрагментированных пакетов;
- port-scan – защита от port scan атак;
- ip-sweep – защиту от IP-sweep атак;
- syn-fin – блокировка TCP-пакетов, с установленными флагами SYN и FIN;
- fin-no-ack – блокировка TCP-пакетов с установленным флагом FIN и не установленным флагом ACK;
- no-flag – блокировка TCP-пакетов, с нулевым полем flags;
- spoofing – защита от IP spoofing атак;
- reserved – блокировка всех ICMP-пакетов 2 и 7 типов (reserved);
- quench – блокировка всех ICMP-пакетов 4 типа (source quench);
- echo-request – блокировка всех ICMP пакетов 8 типа (echo-request);
- time-exceeded – блокировка всех ICMP-пакетов 11 типа (time exceeded);
- unreachable – блокировка всех ICMP-пакетов 3 типа (destination-unreachable);
- icmp-large – блокировка ICMP-пакетов большого объема;
- tcp-all-flags – блокировка tcp-пакетов с флагами;
- udp-frag – блокировка udp-пакетов с флагами.

Без указания ключа <LINK> – активируется отправка всех трапов данной группы.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# snmp-server enable traps screens reserved
```

## snmp-server enable traps snmp

Данная команда разрешает отправку SNMP-уведомлений об изменении параметров окружения.

Использование отрицательной формы команды (no) отменяет разрешение отправки указанных типов уведомлений.

## Синтаксис

```
[no] snmp-server enable traps snmp [ <ACT> ]
```

## Параметры

<ACT> – типы фильтров параметров окружения:

- authentication – уведомления о snmp-запросах на маршрутизатор с неверными community или snmpv3-паролем;
- coldstart – уведомления о перезапуске snmp-сервера на маршрутизаторе;
- linkdown – информация о изменении состояния link в down;
- linkup – информация о изменении состояния link в up.

Без указания ключа <ACT> – активируется отправка всех трапов данной группы.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# snmp-server enable traps snmp linkup
```

## snmp-server enable traps syslog

Данная команда разрешает отправку SNMP-уведомлений с syslog-сообщениями.

Использование отрицательной формы команды (no) отменяет разрешение отправки указанных типов уведомлений.

### Синтаксис

```
[no] snmp-server enable traps syslog
```

### Параметры

Отсутствуют.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# snmp-server enable traps syslog
```

### snmp-server enable traps wifi

Данная команда разрешает отправку SNMP-уведомлений с сообщениями о работе Wi-Fi контроллера. Использование отрицательной формы команды (no) отменяет разрешение отправки указанных типов уведомлений.

### Синтаксис

```
[no] snmp-server enable traps wifi [ <NAME> ]
```

### Параметры

<NAME> – типы трапа о туннелях softgre:

- wifi-tunnels-number-in-bridge-high – включение трапов о превышении количества sub-gre-туннелей включенных в bridge;
- wifi-tunnels-operation – включение трапов о результате snmp-операций с туннелями softgre.

Без указания ключа <NAME> – активируется отправка всех трапов данной группы.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# snmp-server enable traps syslog
```

## snmp-server host

Данной командой включается передача SNMP-уведомлений на указанный IP-адрес и осуществляется переход в режим настройки SNMP-уведомлений.

Использование отрицательной формы команды (no) отключает передачу уведомлений на указанный коллектор SNMP-уведомлений.

### Синтаксис

```
[no] snmp-server host { <ADDR> | <IPV6-ADDR> } [vrf <VRF>]
```

### Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<VRF> – имя экземпляра VRF, в котором находится коллектор SNMP-уведомлений, задается строкой до 31 символа.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

### Пример

```
esr(config)# snmp host 192.168.2.2
```

## snmp-server location

Данной командой устанавливается значение переменной SNMP, содержащей информацию о расположении оборудования (по умолчанию не определено). Для удобства в параметрах можно указать город, улицу, район, номер комнаты и т.п.

Использование отрицательной формы команды (no) удаляет значение переменной, содержащей информацию о расположении оборудования.

### Синтаксис

```
[no] snmp-server location <LOCATION>
```

### Параметры

<LOCATION> – информация о расположении оборудования, задаётся строкой до 255 символов.

### Необходимый уровень привилегий

15



## Командный режим

CONFIG

### Пример

```
esr(config)# snmp-server location doug_lasa_adamsa_42
```

### snmp-server system-shutdown

Данной командой разрешается перезагрузка маршрутизатора при помощи SNMP-сообщений.

Использование отрицательной формы команды (no) запрещает перезагрузку маршрутизатора при помощи SNMP-сообщений.

### Синтаксис

```
[no] snmp-server system-shutdown
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

15

## Командный режим

CONFIG

### Пример

```
esr(config)# snmp-server system-shutdown
```

### snmp-server trap link

Данной командой устанавливается режим отправки SNMP-trap.

Использование отрицательной формы команды (no) устанавливает режим по умолчанию.

### Синтаксис

```
snmp-server trap link <MODE>
```

```
no snmp-server host
```

## Параметры

<MODE> – режим отправки SNMP-trap. Принимает значения:

- ietf;
- cisco.

## Значение по умолчанию

ietf

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# snmp-server trap link cisco
```

## snmp-server user

Данной командой создается SNMPv3-пользователь.

Использование отрицательной формы команды (no) удаляет SNMPv3-пользователя.

## Синтаксис

```
[no] snmp-server user <NAME>
```

## Параметры

<NAME> – имя пользователя, задаётся строкой от 1 до 128 символов.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# snmp-server user admin  
esr(config-sntp-user)#
```

**snmp-server view**

Данной командой создается профиль snmp view, позволяющий разрешать или запрещать доступ к тем или иным OID для community (SNMPv2) и user (SNMPv3).

Использование отрицательной формы команды (no) удаляет профиль snmp view.

**Синтаксис**

```
[no] snmp-server view <VIEW-NAME>
```

**Параметры**

<VIEW-NAME> – имя профиля SNMP view, задаётся строкой до 31 символа.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# snmp-server view user_access
esr(config-snmp-view)#
```

**source-address**

Данной командой определяется IP-адрес для отправки уведомлений на удаленный сервер.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
source-address { <ADDR> | <IPV6-ADDR> | object-group <NETWORK_OBJ_GROUP_NAME> }
no source-address
```

**Параметры**

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

<NETWORK\_OBJ\_GROUP\_NAME> – список адресов, которые будут использоваться в качестве source address.

**Значение по умолчанию**

IPv4/IPv6 – адрес интерфейса, ближайшего к удаленному SNMP-серверу.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-SNMP-HOST

## Пример

```
esr(config-snmp-host)# source-address 192.168.22.17
```

## source-interface

Данной командой определяется интерфейс или туннель маршрутизатора, IPv4/IPv6-адрес которого будет использоваться для отправки уведомлений на удаленный сервер.

Использование отрицательной формы команды (no) удаляет указанный интерфейс или туннель.

## Синтаксис

```
source-interface { <IF> | <TUN> }
```

```
no source-interface
```

## Параметры

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#).

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-SNMP-HOST

## Пример

```
esr(config-snmp-host)# source-interface gigabitethernet 1/0/1
```

## view

Данной командой устанавливается профиль snmp view, позволяющий разрешать или запрещать доступ к тем или иным OID для SNMPv3 user.

Использование отрицательной формы команды (no) удаляет профиль snmp view.

**Синтаксис**

```
[no] view <VIEW-NAME>
```

**Параметры**

<VIEW-NAME> – имя SNMP view профиля, на основании которого обеспечивается доступ к OID, задается строкой до 31 символа.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-SNMP-USER

**Пример**

```
esr(config-snmp-user)# view user_view
```

## Управление SYSLOG

- [logging aaa configuration](#)
- [logging acl configuration](#)
- [logging firewall configuration](#)
- [logging login on-failure](#)
- [logging nat](#)
- [logging service start-stop](#)
- [logging syslog configuration](#)
- [logging userinfo](#)
- [match process-name](#)
- [ntp logging](#)
- [port](#)
- [remote-address](#)
- [severity](#)
- [source-address](#)
- [show syslog](#)
- [show syslog configuration](#)
- [syslog cli-commands](#)
- [syslog console](#)
- [syslog file](#)
- [syslog file-size](#)
- [syslog host](#)
- [syslog max-files](#)
- [syslog monitor](#)
- [syslog program-name](#)
- [syslog reload debugging](#)
- [syslog sequence-numbers](#)
- [syslog snmp](#)
- [syslog timestamp msec](#)
- [syslog sip-level](#)
- [transport](#)
- [vrf](#)

### logging aaa configuration

Данной командой включается запись в локальный syslog-сервер сообщений о изменении конфигурации разделов AAA.

Использование отрицательной формы команды (no) выключает логирование сообщений о изменении конфигурации разделов AAA.

### Синтаксис

```
[no] logging aaa configuration
```

### Параметры

Отсутствуют.

### Значение по умолчанию

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# logging aaa configuration
```

**logging acl configuration**

Данной командой включается запись в локальный syslog-сервер сообщений о изменении конфигурации ACL.

Использование отрицательной формы команды (no) выключает логирование сообщений о изменении конфигурации ACL.

**Синтаксис**

```
[no] logging acl configuration
```

**Параметры**

Отсутствуют.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# logging acl configuration
```

**logging firewall configuration**

Данной командой включается запись в локальный syslog-сервер сообщений о изменении конфигурации межсетевое экрана.

Использование отрицательной формы команды (no) выключает логирование сообщений о изменении конфигурации межсетевое экрана.

**Синтаксис**

```
[no] logging firewall configuration
```

**Параметры**

Отсутствуют.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# logging firewall configuration
```

**logging login on-failure**

Данной командой включается запись в локальный syslog-сервер сообщений о неуспешных попытках подключения к CLI.

Использование отрицательной формы команды (no) выключает логирование неудачных подключений.

**Синтаксис**

```
[no] logging login on-failure
```

**Параметры**

Отсутствуют.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG



## Пример

```
esr(config)# logging login on-failure
```

## logging nat

Данной командой включается запись в локальный syslog-сервер сообщений о работе сервиса NAT. Использование отрицательной формы команды (no) выключает логирование сервиса NAT.

## Синтаксис

```
[no] logging nat [<NAT-TYPE>]
```

## Параметры

<NAT-TYPE> – тип сервиса NAT подлежащий логированию:

- destination;
- proxy;
- source.

## Значение по умолчанию

Отключено.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# logging nat source
```

## logging service start-stop

Данной командой включается запись в локальный syslog-сервер сообщений о запуске и остановке используемых сервисов.

Использование отрицательной формы команды (no) выключает логирование запуска и остановки используемых сервисов.

## Синтаксис

```
[no] logging service start-stop
```

## Параметры

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# logging service start-stop
```

**logging syslog configuration**

Данной командой включается запись в локальный syslog-сервер сообщений о изменении конфигурации syslog-сервера.

Использование отрицательной формы команды (no) выключает логирование сообщений о изменении конфигурации syslog-сервера.

**Синтаксис**

```
[no] logging syslog configuration
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# logging syslog configuration
```

**logging userinfo**

Данной командой включается запись в локальный syslog-сервер сообщений о изменении user-profile.

Использование отрицательной формы команды (no) выключает логирование сообщений об изменении user-profile.

### Синтаксис

```
[no] logging userinfo
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

### Пример

```
esr(config)# logging userinfo
```

### match process-name

Данной командой активируется или деактивируется сохранение в локальный syslog-файл или отправка на удаленный syslog-сервер событий работы отдельных процессов маршрутизатора.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
match [not] process-name <PROCESS-NAME>
no match process-name { <PROCESS-NAME> | all }
```

### Параметры

<PROCESS-NAME> – имя процесса события, которое необходимо или не нужно сохранять в локальный syslog-файл или отправлять на удаленный syslog-сервер. Принимает значения:

- aaa-mgr – вывод информационных сообщений работы сервисов AAA на устройстве;
- alarm-mgr – вывод информационных сообщений менеджера аварийных событий устройства;
- avahi – вывод информационных сообщений сервиса mDNS;
- aux – вывод информационных сообщений работы менеджера, передающего данные из последовательного порта в систему (только на ESR-21/31);
- cfgsync-mgr – вывод информационных сообщений менеджера конфигурации устройства (применения/отката/сохранения/загрузки);
- cli – вывод информационных сообщений командной оболочки CLI;
- sr-mgr – вывод информационных сообщений менеджера запуска и контроля процесса маршрутизации;

- device-mgr – вывод информационных сообщений подключаемых устройств (USB-flash,mmc,SD,HDD);
- dhcp-client – вывод информационных сообщений DHCP-клиента;
- dhcp-proxy – вывод информационных сообщений DHCP-проxy;
- dhcp-server – вывод информационных сообщений DHCP-сервера;
- dns – вывод информационных сообщений сервиса DNS;
- e1d – вывод информационных сообщений E1/Multilink;
- env-mgr – вывод информационных сообщений мониторинга аппаратных параметров окружения системы (БП/FAN/RAM/FLASH/CPU);
- esrfs – вывод информационных сообщений, связанных с файловой системой устройства (чтение, запись);
- file-mgr – вывод информационных сообщений при работе с файлами (операция copy);
- firewall-failover – вывод информационных сообщений сервиса Firewall-failover;
- firewalld – вывод информационных сообщений сервиса Firewall;
- haveged – вывод информационных сообщений службы генератора случайных чисел устройства;
- if-mgr – вывод информационных сообщений менеджера интерфейсов (состояние, сбор статистики);
- if-mgr-ng – вывод информационных сообщений менеджера интерфейсов (конфигурирование и перенастройка);
- ipc-hub – вывод информационных сообщений межпроцессного взаимодействия между различными сервисами системы;
- ipc-mgr – логирование событий IPC-шины устройства;
- ips – вывод информационных сообщений отработки правил IPS/IDS;
- ips-storage-mgr – вывод информационных сообщений хранения IPS/IDS-сигнатур;
- ipsec – вывод информационных сообщений сервиса IPsec;
- key-safed – вывод информационных сообщений сервиса безопасного хранения криптоключей на внешних носителях и подключения/отключения криптохранилищ на внешних носителях;
- l2tpclient – вывод информационных сообщений L2TP-клиента;
- lb – вывод информационных сообщений сервиса балансировки трафика на CPU;
- lic-mgr – вывод информационных сообщений сервиса лицензирования;
- lldpd – вывод информационных сообщений LLDP;
- modem-cfgmgr – вывод информационных сообщений конфигурирования (параметры, применение, конфликт конфигурации) USB-модемов на маршрутизаторе;
- modem-mgr – логирование низкоуровневых процессов подключения USB-модемов к устройству (идентификация типа и дальнейшее подключение);
- multiwan – вывод информационных сообщений MultiWAN;
- nhrp – вывод информационных сообщений NHRP;
- ntp – вывод информационных сообщений NTP;
- oi-mgr – вывод информационных сообщений менеджера оперативной информации внутрисистемных процессов устройства;
- openvpn – вывод информационных сообщений OpenVPN;
- pbx - вывод информационных сообщений внутреннего PBX на VoIP-моделях устройств (только на ESR-12/15(VF));
- voip - вывод информационных сообщений VoIP (только на ESR-12/15(VF));
- pci-boot - вывод информационных сообщений взаимодействия двух процессоров при загрузке firmware на устройство (только на ESR-1700);
- pppoe – вывод информационных сообщений PPPoE;
- pptp – вывод информационных сообщений PPTP;
- prober – вывод информационных сообщений сервиса bypass (только на ESR-10 SLA);
- gexd – вывод информационных сообщений работы регулярных выражений (компиляция, проверка на совпадения и отработка);
- rngd – вывод информационных сообщений службы генератора случайных чисел устройства;
- routing – вывод информационных сообщений процесса маршрутизации;
- serial-iface – вывод информационных сообщений работы менеджера, обрабатывающего данные, полученные из последовательного порта в систему (только на ESR-21/31);
- service-mgr – вывод информационных сообщений менеджера вторичных сервисов устройства;

- session-mgr – вывод информационных сообщений менеджера регистрации и мониторинга CLI-сессий;
- sflowd – вывод информационных сообщений сервиса sFlow;
- sla-mgr – вывод информационных сообщений работы тестов IP SLA;
- snmp-server – вывод информационных сообщений SNMP;
- ssh-server – вывод информационных сообщений SSH-сервера;
- switch – вывод информационных сообщений работы аппаратного чипа коммутации (только на ESR-1000);
- switch-ng – вывод информационных сообщений работы аппаратного чипа коммутации (только на ESR-1200/1500/1511/1700);
- sync-mgr – вывод информационных сообщений синхронизации служебных данных системы на удалённых устройствах;
- syslog-mgr – вывод информационных сообщений менеджера логирования (выборка процессов логирования, классифицирование по необходимому уровню логирования);
- syslog-ng – вывод информационных сообщений менеджера логирования (отправка сгенерированных сообщений в необходимый интерфейс вывода данных);
- systemdb – вывод информационных сообщений менеджера аварий сервисов системы;
- telnet-server – вывод информационных сообщений Telnet-сервера;
- vrrp – вывод информационных сообщений VRRP;
- zabbix-agent – вывод информационных сообщений Zabbix-агента;
- zabbix-proxy – вывод информационных сообщений Zabbix-прокси сервера;
- others – логирование прочих процессов, не попадающих под какую-либо классификацию.

all – ключ для удаление всех ранее сконфигурированных процессов.

### Значение по умолчанию

Сохраняются/отправляются события всех процессов.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-SYSLOG-SNMP

CONFIG-SYSLOG-HOST

CONFIG-SYSLOG-FILE

### Пример

```
esr(config-syslog-file)# match not process-name dhcp-client
```

### ntp logging

Данной командой включается запись в локальный syslog-сервер сообщений о работе с NTP-пирами.

Использование отрицательной формы команды (no) выключает логирование сообщений о работе с NTP-пирами.

### Синтаксис

[no] ntp logging

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Отключено.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# ntp logging
```

## port

Данной командой указывается номер TCP/UDP-порта, на который будут отправляться пакеты с syslog-сообщениями.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
port <PORT>
```

```
no port
```

## Параметры

<PORT> – номер TCP/UDP-порта, на который будут отправляться пакеты с syslog-сообщениями.

## Значение по умолчанию

514

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-SYSLOG-HOST

## Пример

```
esr(config-syslog-host)# port 5014
```

## remote-address

Данной командой указывается IPv4/IPv6-адрес удаленного syslog-сервера.

Использование отрицательной формы команды (no) удаляет IPv4/IPv6-адрес удаленного syslog-сервера.

### Синтаксис

```
remote-address { <ADDR> | <IPV6-ADDR> }  
no remote-address
```

### Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-SYSLOG-HOST

### Пример

```
esr(config-syslog-host)# remote-address 192.168.0.100
```

## severity

Данной командой указывается уровень важности сообщений, которые будут сохраняться в локальный syslog-файл или отправляться на удаленный syslog-сервер.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
severity <SEVERITY>  
no severity
```

### Параметры

<SEVERITY> – уровень важности сообщения, принимает значения (в порядке убывания важности):

- none – отключает вывод syslog-сообщений в консоль;
- emerg – в системе произошла критическая ошибка, система неработоспособна;
- alert – сигналы тревоги, необходимо немедленное вмешательство персонала;
- crit – критическое состояние системы, сообщение о событии;
- error – сообщения об ошибках;

- warning – предупреждения, неаварийные сообщения;
- notice – сообщения о важных системных событиях;
- info – информационные сообщения системы;
- debug – отладочные сообщения, предоставляют пользователю информацию для корректной настройки системы.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-SYSLOG-SNMP

CONFIG-SYSLOG-HOST

CONFIG-SYSLOG-FILE

### Пример

```
esr(config-syslog-file)# severity info
```

### source-address

Данной командой указывается IPv4/IPv6-адрес маршрутизатора, от которого будут отправляться пакеты на удаленный syslog-сервер.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
source-address { <ADDR> | <IPV6-ADDR> | object-group <NETWORK_OBJ_GROUP_NAME> }
```

```
no source-address
```

### Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

<NETWORK\_OBJ\_GROUP\_NAME> – список адресов, которые будут использоваться в качестве source address.

### Значение по умолчанию

IPv4/IPv6-адрес интерфейса, с которого отправляются пакеты на удаленный syslog-сервер.

### Необходимый уровень привилегий

10



**Командный режим**

CONFIG-SYSLOG-HOST

**Пример**

```
esr(config-syslog-host)# source-address 192.168.54.157
```

**show syslog**

Команда для просмотра текущей информации о конфигурации syslog-журнала, списка созданных log-файлов, а также для просмотра log-файлов с возможностью фильтрации с помощью регулярных выражений.

**Синтаксис**

```
show syslog <FILE> [ from-date <YEAR> <MONTH> <DAY> ] [ from-time <TIME> ] [ to-date <YEAR> <MONTH> <DAY> ] [ to-time <TIME> ] [ from-end ]
```

**Параметры**

<FILE> – имя файла, задаётся строкой до 31 символа;

from-date – для вывода информации, начиная с указываемой даты;

from-time – для вывода информации, начиная с указываемого времени;

to-date – для вывода информации до указываемой даты;

to-time – для вывода информации до указываемого времени;

<YEAR> – год, принимает значения [2001..2037].

<MONTH> – месяц, принимает значения [ January/February/March/April/May/June/July/August/September/October/November/December ];

<DAY> – день месяца, принимает значения [1..31];

<TIME> – устанавливаемое системное время, задаётся в виде HH MM SS, где:

- HH – часы, принимает значение [0..23];
- MM – минуты, принимает значение [0 .. 59];
- SS – секунды, принимает значение [0 .. 59].

from-end – просмотр содержимого файла с конца, так как последние записи помещаются в конец файла.

**Необходимый уровень привилегий**

15

**Командный режим**

ROOT

**Пример**

```
esr# show syslog
Name                               Size                               Date of last modification
```

```

-----
flash:critlog/log                113.48   KB   2024-06-27 10:14:50
flash:critlog/log.1             499.99   KB   2023-12-17 02:33:04
flash:critlog/reload_reason.txt  78.00    B    2024-06-06 10:01:27
tmpsys:syslog/auth.log          62.11   KB   2024-07-02 08:33:20
Total files: 4

```

```
esr# show syslog configuration
```

```
SYSLOG
```

```
File size: 512 (kiB)
```

```
Number of logs: 3
```

```
Console: info
```

```
Files:
```

```
~~~~~
```

ID	Name	Severity
0	esr	info

## show syslog configuration

Команда для просмотра текущей информации о конфигурации syslog-журнала.

### Синтаксис

```
show syslog configuration [ console | files [<FILE>] | firewall | hosts [<NAME>] |
monitor | snmp ]
```

### Параметры

console – конфигурация syslog-событий при консольном подключении;

files – информация о файлах для записи syslog-событий;

firewall – конфигурация syslog-событий межсетевого экрана;

hosts – вывод информации о хостах, на которые передаются syslog-журналы;

monitor – вывод конфигурации монитора syslog-журнала;

snmp – вывод конфигурации snmp syslog-журнала.

### Необходимый уровень привилегий

15

### Командный режим

ROOT

### Пример

```

esr# show syslog configuration
Syslog:
  Write sequence numbers:      Disabled
  Write msec in timestamp:     Disabled
  Write program names:        Disabled

  Log syslog configuration changes: Disabled
  Debug severity on reload:    Disabled

```

```

Console:
  Console severity:          info
  Console filters defined:   Disabled

```

```

Monitor:
  Monitor severity:         info
  Monitor filters defined:   Disabled

```

```

User Info:
  Log AAA configuration changes: Disabled
  Log ACL configuration changes: Disabled
  Log user info changes:     Disabled
  Log user commands:         Disabled
  Log failed authentication attempts: Disabled

```

```

IPS:
  IPS severity:             info

```

```

SNMP:
  SNMP trap status:         Disabled
  SNMP severity:            info
  SNMP filters defined:     Disabled

```

```

Network:
  Log IPv4 BFD neighbor changes: Disabled
  Log IPv4 BGP neighbor changes: Disabled
  Log IPv4 OSFP neighbor changes: Disabled

  Log IPv6 BFD neighbor changes: Disabled
  Log IPv6 BGP neighbor changes: Disabled
  Log IPv6 OSFP neighbor changes: Disabled

  Log NAT source:           Disabled
  Log NAT destination:      Disabled
  Log HTTP proxy:           Disabled

  NTP peer logging:         Disabled

```

```

SLA:
  Log SLA delay:            Disabled
  Log SLA error:            Disabled
  Log SLA jitter:           Disabled
  Log SLA losses:           Disabled
  Log SLA status:           Disabled

```

```

Firewall:
  Log firewall configuration changes: Disabled
  Log firewall screen:       Disabled
  Detailed firewall screen logging: Disabled

```

#### Files:

~~~~~

| Name                   | Severity | Filters |
|------------------------|----------|---------|
| tmpsys:syslog/DHCP     | Info     | Yes     |
| tmpsys:syslog/debug    | Debug    | No      |
| tmpsys:syslog/NOT-DHCP | Info     | Yes     |
| tmpsys:syslog/test     | Debug    | Yes     |

#### Hosts:

~~~~~

Name	IP Address	Severity	Transport	Port	Filters
------	------------	----------	-----------	------	---------

syslog_host	192.168.1.2	Info	UDP	514	No
-------------	-------------	------	-----	-----	----

### syslog cli-commands

Данной командой включается процесс логирования введённых команд пользователя на локальный syslog-сервер.

Использование отрицательной формы команды (no) выключает логирование команд.

#### Синтаксис

```
[no] syslog cli-commands
```

#### Параметры

Команда не содержит параметров.

#### Необходимый уровень привилегий

15

#### Командный режим

CONFIG

#### Пример

```
esr(config)# syslog cli-commands
```

### syslog console

Данной командой включается отображение syslog-сообщений при консольном подключении и происходит переход в режим конфигурирования параметров.

Использование отрицательной формы команды (no) устанавливает отображение сообщений по умолчанию.

#### Синтаксис

```
syslog console
no syslog console
```

#### Параметры

Команда не содержит параметров.

#### Необходимый уровень привилегий

15

## Командный режим

CONFIG

### Пример

```
esr(config)# syslog console
```

### syslog file

Данной командой включается сохранение сообщений syslog в указанный файл и происходит переход в режим конфигурирования параметров данного syslog-файла.

Использование отрицательной формы команды (no) отключает сохранение сообщений syslog в указанный файл.

### Синтаксис

```
syslog file { flash:syslog/<NAME> | tmpsys:syslog/<NAME> }  
no syslog file { flash:syslog/<NAME> | tmpsys:syslog/<NAME> | all}
```

### Параметры

flash – файл располагается в энергонезависимой памяти устройства;

tmpsys – файл располагается в энергозависимой памяти устройства;

<NAME> – имя файла, в который будет производиться запись сообщений заданного уровня, задается строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут отключено сохранение во все сконфигурированные syslog-файлы.

- ❌ Для моделей ESR-1x размер файловой системы tmpsys составляет 50 МБ, для всех остальных моделей – 100 МБ.  
Настраивая число файлов в ротации и размер файла, необходимо следить за тем, чтобы суммарный размер файлов не превышал эти ограничения.

### Необходимый уровень привилегий

15

## Командный режим

CONFIG

### Пример

```
esr(config)# syslog file flash:syslog/file
```

### syslog file-size

Командой устанавливается максимальный размер файла журнала. По превышении указанного размера будет автоматически производиться ротация файлов.

Использование отрицательной формы команды (no) устанавливает значение размера файла журнала в значение по умолчанию.

### Синтаксис

```
syslog file-size <SIZE>  
no syslog file-size
```

### Параметры

<SIZE> – размер файла, принимает значение [10..10000000] Кбайт.

### Значение по умолчанию

500 Кбайт.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

### Пример

```
esr(config)# syslog file-size 10000
```

### syslog host

Данной командой включается передача сообщений syslog на удаленный syslog-сервер и происходит переход в режим конфигурирования параметров данного syslog-сервера.

Использование отрицательной формы команды (no) отключает передачу сообщений syslog на удаленный syslog-сервер.

### Синтаксис

```
syslog host <HOSTNAME>  
no syslog host { <HOSTNAME> | all }
```

### Параметры

<HOSTNAME> – наименование syslog-сервера, задаётся строкой до 31 символа. Используется только для идентификации сервера при конфигурировании. Значение «all» используется в команде **no syslog host all** для удаления всех syslog-серверов.

### Необходимый уровень привилегий

15

## Командный режим

CONFIG

### Пример

```
esr(config)# syslog host eltex
```

### syslog max-files

Данная команда устанавливает максимальное количество файлов, сохраняемых при ротации.

Использование отрицательной формы команды (no) устанавливает количество хранимых файлов журнала в значение по умолчанию.

### Синтаксис

```
syslog max-files <NUM>
```

```
no syslog max-files
```

### Параметры

<NUM> – максимальное количество файлов, принимает значения [1 .. 1000].

### Значение по умолчанию

15

### Необходимый уровень привилегий

15

## Командный режим

CONFIG

### Пример

```
esr(config)# syslog max-files 100
```

### syslog monitor

Данной командой включается отображение syslog-сообщений при удаленных подключениях (Telnet, SSH) и происходит переход в режим конфигурирования параметров.

Использование отрицательной формы команды (no) устанавливает отображение сообщений по умолчанию.

### Синтаксис

```
syslog monitor
```

```
no syslog monitor
```

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# syslog monitor
```

## syslog program-name

Данной командой включается добавление имени процесса к записям в локальный syslog-сервер. Использование отрицательной формы команды (no) убирает имя процесса из сообщений.

## Синтаксис

```
syslog program-name  
no syslog program-name
```

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# program-name
```

## syslog reload debugging

Данной командой включается режим debugging в процессе перезагрузки для локального syslog-сервера. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
[no] syslog reload debugging
```



**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# syslog reload debugging
```

**syslog sequence-numbers**

Данной командой включается добавление порядкового номера к записям в локальный syslog-сервер. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
[no] syslog sequence-numbers
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# syslog sequence-numbers
```

## syslog snmp

Данной командой включается отображение syslog-сообщений на snmp-сервер в виде snmp-trap и происходит переход в режим конфигурирования параметров.

Использование отрицательной формы команды (no) устанавливает отправку сообщений по умолчанию.

### Синтаксис

```
syslog snmp  
no syslog snmp
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

10


### Командный режим

CONFIG

### Пример

```
esr(config)# syslog snmp
```

## syslog timestamp msec

 В данной версии ПО команда применима только для маршрутизаторов ESR-100/200/1000/1200/1500/1511/1700/3100/3200.

Данной командой включается добавление миллисекунд в таймштамп.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
[no] syslog timestamp msec
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

15

## Командный режим

CONFIG

### Пример

```
esr(config)# syslog timestamp msec
```

### syslog sip-level

Данной командой устанавливается уровень логирования sip-событий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
[no] syslog sip-level <LEVEL>
```

### Параметры

<LEVEL> – уровень sip-сообщений, который будут попадать в syslog. Принимает значение в диапазоне [0..9].

### Значение по умолчанию

0

### Необходимый уровень привилегий

15

## Командный режим

CONFIG

### Пример

```
esr(config)# syslog sip-level 8
```

### transport

Данной командой указывается транспортный протокол для передачи пакетов на удаленный syslog-сервер.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
transport { tcp | udp }  
no transport
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

udp

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-SYSLOG-HOST

## Пример

```
esr(config-syslog-host)# transport tcp
```

## vrf

Данной командой указывается имя экземпляра VRF, в рамках которого будут отправляться пакеты на удаленный syslog-сервер.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
vrf <VRF>
```

```
no vrf
```

## Параметры

<VRF> – имя экземпляра VRF, в котором доступен удаленный syslog-сервер, задается строкой до 31 символа.

## Значение по умолчанию

Отсутствует (глобальная таблица маршрутизации).

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-SYSLOG-HOST

## Пример

```
esr(config-syslog-host)# vrf MGMT
```

## Настройка доступа SSH, Telnet

- `ip ftp client password`
- `ip ftp client username`
- `ip ftp source-ip`
- `ip sftp client password`
- `ip sftp client username`
- `ip ssh access-addresses`
- `ip ssh authentication algorithm disable`
- `ip ssh authentication retries`
- `ip ssh authentication timeout`
- `ip ssh client password`
- `ip ssh client source-ip`
- `ip ssh client username`
- `ip ssh dscp`
- `ip ssh encryption algorithm disable`
- `ip ssh host-key algorithm`
- `ip ssh key-exchange algorithm disable`
- `ip ssh key-exchange time`
- `ip ssh key-exchange volume`
- `ip ssh port`
- `ip ssh server`
- `ip telnet access-addresses`
- `ip telnet dscp`
- `ip telnet port`
- `ip telnet server`
- `ip tftp client timeout`
- `show crypto key mypubkey`
- `update ssh-host-key`

### ip ftp client password

Данной командой определяется пароль по умолчанию для операций копирования по протоколу FTP. Использование отрицательной формы команды (no) удаляет пароль.

### Синтаксис

```
ip ftp client password { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
[no] ftp client password
```

### Параметры

<CLEAR-TEXT> – пароль, задаётся строкой [1..16] символов, принимает значения [0-9a-fA-F];

<ENCRYPTED-TEXT> – зашифрованный пароль, задаётся строкой [2..32] символов.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

## Пример

```
esr(config)# ip ftp client password test
```

## ip ftp client username

Данной командой определяется имя пользователя по умолчанию для операций копирования по протоколу FTP.

Использование отрицательной формы команды (no) удаляет имя пользователя.

## Синтаксис

```
ip ftp client username <NAME>
```

```
no ftp client username
```

## Параметры

<NAME> – имя пользователя, задаётся строкой до 31 символа.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# ip ftp client username test
```

## ip ftp source-ip

Данной командой указывается адрес, с которого будет устанавливаться соединение с FTP-сервером.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
ip ftp source-ip <ADDR>
```

```
[no] ip ftp source-ip
```

## Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

## Значение по умолчанию

IP-адрес интерфейса, с которого отправляются пакеты на FTP-сервер.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip ftp source-ip 192.168.1.57
```

**ip sftp client password**

Данная команда задает значения пароля пользователя для клиента протокола SFTP.

Использование отрицательной формы команды (no) удаляет значения пароля пользователя для клиента протокола SFTP.

**Синтаксис**

```
ip sftp client password { <TEXT> | encrypted < ENCRYPTED-TEXT > }
```

```
no ip sftp client password
```

**Параметры**

<TEXT> – строка [1..16] ASCII-символов;

<ENCRYPTED-TEXT> – зашифрованный пароль, размером [8..16] байт, задается строкой [16..32].

**Значение по умолчанию**

Имя пользователя не задано.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip sftp client password 123456789
```

**ip sftp client username**

Данная команда задает значения имени пользователя для клиента протокола SFTP.

Использование отрицательной формы команды (no) удаляет значения имени пользователя для клиента протокола SFTP.

**Синтаксис**

```
ip sftp client username <USERNAME>
no ip sftp client username
```

**Параметры**

<USERNAME> – имя пользователя, задаётся строкой до 31 символа.

**Значение по умолчанию**

Имя пользователя не задано.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG


**Пример**

```
esr(config)# ip sftp client username esruser
```

**ip ssh access-addresses**

Данная команда ограничивает доступ до SSH-сервера. SSH-сервер становится доступным только с определённых адресов.

Использование отрицательной формы команды (no) разрешает доступ с любых адресов.

 Данная команда позволяет ограничивать доступ до SSH-сервера при отключенных функциях Firewall или ACL. В случае совместного использования данного функционала с функциями Firewall и ACL необходимо, чтобы трафик был разрешён всеми тремя функциями.

**Синтаксис**

```
ip ssh access-addresses [ vrf <NAME> ] <OBJ-GR-NAME>
no ip ssh access-addresses [ vrf <NAME> ]
```

**Параметры**

<OBJ-GR-NAME> – имя профиля IP-адресов, с которых разрешён доступ.

<VRF> – имя экземпляра VRF, в рамках которого будет работать ограничение доступа на SSH-сервер.

**Необходимый уровень привилегий**

15



**Значение по умолчанию**

Доступ разрешён с любых адресов.

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip ssh access-addresses MGT
```

**ip ssh authentication algorithm disable**

Данная команда запрещает использование определенного алгоритма аутентификации для SSH-сервера.

Использование отрицательной формы команды (no) разрешает использование определенного алгоритма аутентификации для SSH-сервера.

**Синтаксис**

```
[no] ip ssh authentication algorithm <ALGORITHM> disable
```

**Параметры**

<ALGORITHM> – алгоритм хеширования, принимает значения [md5, md5-96, sha1, sha1-96, sha2-256, sha2-512, ripemd160].

**Необходимый уровень привилегий**

15

**Значение по умолчанию**

Разрешены все алгоритмы аутентификации.

**Командный режим**

CONFIG

**Пример**

```
esr(config)# no ip ssh authentication algorithm md5 disable
```

**ip ssh authentication retries**

Данная команда устанавливает количество попыток аутентификации для SSH-сервера.

Использование отрицательной формы команды (no) устанавливает количество попыток аутентификации для SSH-сервера по умолчанию.

**Синтаксис**

```
ip ssh authentication retries <NUM>  
no ip ssh authentication retries
```

**Параметры**

<NUM> – количество попыток аутентификации для SSH-сервера [1..10].

**Необходимый уровень привилегий**

10

**Значение по умолчанию**

6

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip ssh authentication retries 5
```

**ip ssh authentication timeout**

Данная команда устанавливает для ssh-сервера на маршрутизаторе время ожидания ввода пароля при аутентификации SSH-клиента.

Использование отрицательной формы команды (no) устанавливает период времени ожидания аутентификации для SSH-сервера по умолчанию.

**Синтаксис**

```
ip ssh authentication timeout <SEC>  
no ip ssh authentication timeout
```

**Параметры**

<SEC> – период времени в секундах, принимает значения [30..360].

**Необходимый уровень привилегий**

10

**Значение по умолчанию**

120

**Командный режим**

CONFIG

## Пример

```
esr(config)# ip ssh authentication timeout 60
```

## ip ssh client password

Данной командой определяется пароль по умолчанию для операций копирования по протоколу SCP. Использование отрицательной формы команды (no) удаляет пароль.

## Синтаксис

```
ip ssh client password { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }  
no ip ssh client password
```

## Параметры

<CLEAR-TEXT> – пароль, задаётся строкой [1..16] символов, принимает значения [0-9a-fA-F];

<ENCRYPTED-TEXT > – зашифрованный пароль, задаётся строкой [2..32] символов.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# ip ssh client password test132
```

## ip ssh client source-ip

Данной командой определяется IP-адрес маршрутизатора, от которого будут устанавливаться SSH-сессии на другие устройства.

Использование отрицательной формы команды (no) удаляет имя пользователя.

## Синтаксис

```
ip ssh client source-ip <ADDR>  
no ip ssh client source-ip
```

## Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Указываемый IP-адрес должен быть назначен на каком-либо интерфейсе/туннеле маршрутизатора.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip ssh client source-ipt 192.168.22.78
```

**ip ssh client username**

Данной командой определяется имя пользователя по умолчанию для операций копирования по протоколу SCP.

Использование отрицательной формы команды (no) удаляет имя пользователя.

**Синтаксис**

```
ip ssh client username <NAME>
```

```
no ip ssh client username
```

**Параметры**

<NAME> – имя пользователя, задаётся строкой до 31 символа.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip ssh client username tester
```

**ip ssh dscp**

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов SSH-сервера.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

**Синтаксис**

```
ip ssh dscp <DSCP>
```

```
no ip ssh dscp
```

## Параметры

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

## Значение по умолчанию

32

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# ip ssh dscp 40
```

## ip ssh encryption algorithm disable

Данная команда запрещает использование определенного алгоритма шифрования для SSH-сервера.

Использование отрицательной формы команды (no) разрешает использование определенного алгоритма шифрования для SSH-сервера.

## Синтаксис

```
[no] ip ssh encryption algorithm <ALGORITHM> disable
```

## Параметры

<ALGORITHM> – идентификатор алгоритма шифрования, принимает значения [aes128, aes192, aes256, aes128ctr, aes192ctr, aes256ctr, arcfour, arcfour128, arcfour256, blowfish, cast128, 3des].

## Необходимый уровень привилегий

15

## Значение по умолчанию

Все алгоритмы разрешены.

## Командный режим

CONFIG

## Пример

```
esr(config)# ip ssh encryption algorithm aes128 disable
```

## ip ssh host-key algorithm

Данная команда запрещает использование определенного алгоритма верификации Host-Key для SSH-сервера.

Использование отрицательной формы команды (no) разрешает использование определенного алгоритма алгоритма верификации Host-Key для SSH-сервера.

### Синтаксис

```
[no] ip ssh host-key algorithm <ALGORITHM> disable
```

### Параметры

<ALGORITHM> – идентификатор алгоритма шифрования, принимает значения [dsa, ecdsa256, ecdsa384, ecdsa521, ed25519, rsa].

### Необходимый уровень привилегий

15

### Значение по умолчанию

Все алгоритмы разрешены.

### Командный режим

CONFIG

### Пример

```
esr(config)# ip ssh host-key algorithm dsa disable
```

## ip ssh key-exchange algorithm disable

Данная команда запрещает использование определенного алгоритма обмена ключами для SSH-сервера.

Использование отрицательной формы команды (no) разрешает использование определенного алгоритма обмена ключами для SSH-сервера.

### Синтаксис

```
[no] ip ssh key-exchange algorithm <ALGORITHM> disable
```

### Параметры

<ALGORITHM> – идентификатор протокола обмена ключами, принимает значения [dh-group1-sha1, dh-group14-sha1, dh-group-exchange-sha1, dh-group-exchange-sha256, ecdh-sha2-nistp256, ecdh-sha2-nistp384, ecdh-sha2-nistp521].

### Необходимый уровень привилегий

15

### Значение по умолчанию

Все алгоритмы разрешены.

### Командный режим

CONFIG

### Пример

```
esr(config)# ip ssh key-exchange algorithm dh-group-exchange-sha1 disable
```

### ip ssh key-exchange time

Данная команда устанавливает период времени смены ключей аутентификации для SSH-сервера.

Использование отрицательной формы команды (no) устанавливает период времени смены ключей аутентификации для SSH-сервера по умолчанию.

### Синтаксис

```
ip ssh key-exchange time <SEC>
```

```
no ip ssh key-exchange time
```

### Параметры

<SEC> – период времени в часах, принимает значения [1..72].

### Необходимый уровень привилегий

15

### Значение по умолчанию

1

### Командный режим

CONFIG

### Пример

```
esr(config)# ip ssh key-exchange time 24
```

### ip ssh key-exchange volume

Данная команда устанавливает объем данных, после прохождения которого произойдет обновление ключей аутентификации для SSH-сервера.

Использование отрицательной формы команды (no) устанавливает объем данных, после прохождения которого произойдет обновление ключей аутентификации для SSH-сервера по умолчанию.

**Синтаксис**

```
ip ssh key-exchange volume <DATA>  
no ip ssh key-exchange volume
```

**Параметры**

<DATA> – объем данных в мегабайтах, принимает значения [1..4096].

**Необходимый уровень привилегий**

15

**Значение по умолчанию**

1000

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip ssh key-exchange volume 512
```

**ip ssh port**

Данной командой определяется порт SSH-сервера на маршрутизаторе.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
ip ssh port <PORT>  
no ip ssh port
```

**Параметры**

<PORT> – номер порта, указывается в диапазоне [1..65535].

**Значение по умолчанию**

22

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG



## Пример

```
esr(config)# ip ssh port 3001
```

## ip ssh server

Данной командой включается SSH-сервер на маршрутизаторе.

Использование отрицательной формы команды (no) отключает SSH-сервер.

## Синтаксис

```
[no] ip ssh server [ vrf <VRF>]
```

## Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать SSH-сервер.

## Значение по умолчанию

SSH-сервер выключен.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG


## Пример

```
esr(config)# no ip ssh server
```

## ip telnet access-addresses

Данная команда ограничивает доступ до Telnet-сервера. Telnet-сервер становится доступным только с определённых адресов.

Использование отрицательной формы команды (no) разрешает доступ с любых адресов.

 Данная команда позволяет ограничивать доступ до Telnet-сервера при отключенных функциях Firewall или ACL.  
В случае совместного использования данного функционала с функциями Firewall и ACL необходимо, чтобы трафик был разрешён всеми тремя функциями.

## Синтаксис

```
ip telnet access-addresses [ vrf <NAME> ] <OBJ-GR-NAME>
```

```
no ip telnet access-addresses [ vrf <NAME> ]
```

## Параметры

<OBJ-GR-NAME> – имя профиля IP-адресов, с которых разрешён доступ.

<VRF> – имя экземпляра VRF, в рамках которого будет работать ограничение доступа на Telnet-сервер.

## Необходимый уровень привилегий

15

## Значение по умолчанию

Доступ разрешён с любых адресов.

## Командный режим

CONFIG

## Пример

```
esr(config)# ip telnet access-addresses MGT
```

## ip telnet dscp

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов Telnet-сервера.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

## Синтаксис

```
ip telnet dscp <DSCP>
```

```
no ip telnet dscp
```

## Параметры

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

## Значение по умолчанию

32

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# ip telnet dscp 40
```

### ip telnet port

Данной командой определяется порт Telnet-сервера на маршрутизаторе.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ip telnet port <PORT>
```

```
no ip telnet port
```

### Параметры

<PORT> – номер порта, принимает значения [1..65535].

### Значение по умолчанию

23

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

## Пример

```
esr(config)# ip telnet port 2001
```

### ip telnet server

Данной командой включается Telnet-сервер на маршрутизаторе.

Использование отрицательной формы команды (no) отключает Telnet-сервер.

### Синтаксис

```
[no] ip telnet server [vrf <VRF>]
```

### Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать Telnet-сервер.

**Значение по умолчанию**

Telnet-сервер выключен.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# no ip telnet server
```

**ip tftp client timeout**

Данной командой настраивается время ожидания ответа от TFTP-сервера для TFTP-клиента на маршрутизаторе.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
ip tftp client timeout <TIMER>  
no ip tftp client timeout
```

**Параметры**

<TIMER> – максимальное значение времени ожидания ответа от TFTP-сервера в секундах, принимает значение [10..60].

**Значение по умолчанию**

10

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip tftp client timeout 15
```

## show crypto key mypubkey

Команда используется для просмотра открытых ключей устройства, используемых при установлении соединения по протоколу SSH.

### Синтаксис

```
show crypto key mypubkey <OPTIONS>
```

### Параметры

<OPTIONS> – алгоритм генерации нового криптографического ключа:

- dsa – алгоритм DSA;
- ecdsa – алгоритм ECDSA. Дополнительно необходимо указать размер ключа, 256, 384 или 521;
- ed25519 – алгоритм ED25519;
- rsa – алгоритм RSA;
- rsa1 – алгоритм RSA1.

### Необходимый уровень привилегий

15

### Командный режим

ROOT

### Пример

```
esr# show crypto key mypubkey rsa
Key data
-----
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQDz750sWCQrnNufg1yhuksTFYCYdEfg
JZ9tWUvcssAZhCJWmewprXBuZMABzFmfBg157pgapxn2qJXJ8ESMV7X7gPfy
xQQah6l376z3SFcpKvwudNgwHiS5HCYPRQWx2Xdaz/nJtYr5NpYgLPba68NC
iXcqEp7EPR5GojDVxpuDuk0hPFcihzmt5Yx8ZptJRzRtsuDQYlowv0Qa24kd
0lQ90/1qKfbAhB6XI60l+dK5VEj7giBESarcRn69/e/YVbdGBdTE93QWFPKI
bm63imfbxRwWtcwsFdIH8Blv9ZqDqqF/I03TkIKa31hV9GnsawLAXi/IdyY
bYPboHRdcTlH/ root@esr-1000
```

## update ssh-host-key

Данной командой генерируется пара новых криптографических ключей для установления соединения по протоколу SSH.

### Синтаксис

```
update ssh-host-key { dsa | escda <ESCDA> | ed25519 <ED25519> | rsa <RSA> }
```

### Параметры

dsa – алгоритм DSA;

ecdsa – алгоритм ECDSA:

- <ECDSA> – размер ключа, принимает значение 256, 384 или 521;
- Без указания используется размер ключа 521.

ed25519 – алгоритм ED25519:

- <ED25519> – размер ключа, принимает значение [256..2048];
- Без указания используется размер ключа 2048.

rsa – алгоритм RSA с указанием длины ключа:

- <RSA> – размер ключа, принимает значение [1024..2048];
- Без указания используется размер ключа 2048.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

### Пример

```
esr(config)# update ssh-host-key ecdsa
```

## Настройка протокола LLDP

- [lldp enable](#)
- [lldp hold-multiplier](#)
- [lldp management-address](#)
- [lldp med fast-start enable](#)
- [lldp med fast-start tx-interval](#)
- [lldp receive](#)
- [lldp system-description](#)
- [lldp system-name](#)
- [lldp timer](#)
- [lldp transmit](#)
- [show lldp neighbors](#)
- [show lldp statistics](#)

### lldp enable

Данной командой на маршрутизаторе включается поддержка протокола LLDP.

Использование отрицательной формы команды (no) отключает поддержку протокола LLDP.

### Синтаксис

```
[no] lldp enable
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

### Пример

```
esr(config)# lldp enable
```

### lldp hold-multiplier

Данной командой устанавливается количество периодов lldp-timer, в течении которых маршрутизатор хранит информацию, полученную по протоколу LLDP.

Использование отрицательной формы команды (no) устанавливает соответствующий параметр в значение по умолчанию.

**Синтаксис**

```
lldp hold-multiplier <SEC>  
no lldp hold-multiplier
```

**Параметры**

<SEC> – период времени в секундах, принимает значение [1..10].

**Значение по умолчанию**

4

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# lldp hold-multiplier 5
```

**lldp management-address**

Данной командой устанавливается IP-адрес, который будет передаваться в LLDP TLV в качестве management-address.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию для поля LLDP TLV management-address.

**Синтаксис**

```
lldp management-address <ADDR>  
no lldp management-address
```

**Параметры**

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Указываемый IP-адрес должен быть назначен на каком-либо интерфейсе/туннеле маршрутизатора.

**Значение по умолчанию**

Один из существующих.

**Необходимый уровень привилегий**

15



## Командный режим

CONFIG

### Пример

```
esr(config)# lldp management-address 192.168.54.42
```

### lldp med fast-start enable

Данной командой включается рассылка LLDP-MED сообщений на интерфейсах с настроенной сетевой политикой (см. раздел [lldp network-policy](#)).

Использование отрицательной формы команды (no) выключает рассылку LLDP-MED сообщений на интерфейсах с настроенной сетевой политикой.

### Синтаксис

```
[no] lldp med fast-start enable
```

### Параметры

Отсутствуют.

### Значение по умолчанию

Рассылка LLDP-MED сообщений отключена.

### Необходимый уровень привилегий

15

## Командный режим

CONFIG

### Пример

```
esr(config)# lldp med fast-start enable
```

### lldp med fast-start tx-interval

Данной командой настраивается период рассылки LLDP-MED сообщений на интерфейсах с настроенной сетевой политикой (см. раздел [lldp network-policy](#)).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию периода рассылки LLDP-MED сообщений на интерфейсах с настроенной сетевой политикой.

### Синтаксис

```
lldp med fast-start tx-interval <SEC>
```

```
[no] lldp med fast-start tx-interval
```

## Параметры

<SEC> – период времени, принимает значения [1..32768] секунд.

## Значение по умолчанию

1 секунда.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# lldp med fast-start tx-interval 20
```

## lldp receive

Данной командой включается обработка LLDPDU на физическом интерфейсе.

Использование отрицательной формы команды (no) отключает обработку LLDPDU на физическом интерфейсе.

## Синтаксис

```
[no] lldp receive
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Прием LLDP-пакетов выключен.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

## Пример

```
esr(config-if-gi)# lldp receive
```

## lldp system-description

Данной командой устанавливается поле, которое будет передаваться в LLDP TLV в качестве system-description.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию для поля LLDP TLV system-description.

## Синтаксис

```
lldp system-description <DESCRIPTION>
```

```
no lldp system-description
```

## Параметры

<DESCRIPTION> – идентификатор описания системы, задаётся строкой до 255 символа.

## Значение по умолчанию

Содержит информацию о модели и версии ПО маршрутизатора.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# lldp system-description esr-1000-1.3.0
```

## lldp system-name

Данной командой устанавливается поле, которое будет передаваться в LLDP TLV в качестве system-name.

Использование отрицательной формы команды (no).

## Синтаксис

```
lldp system-name <NAME>
```

```
no lldp system-name
```

## Параметры

<NAME> – идентификатор имени системы, задаётся строкой до 255 символа.

**Значение по умолчанию**

Совпадает с заданным hostname.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# lldp system-name esr-100-branch-12
```

**lldp timer**

Данной командой устанавливается период отправки LLDPDU.

Использование отрицательной формы команды (no) устанавливает период отправки LLDPDU по умолчанию.

**Синтаксис**

```
lldp timer <SEC>
```

```
no lldp timer
```

**Параметры**

<SEC> – период времени в секундах, принимает значение [1..32768].

**Значение по умолчанию**

30

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# lldp timer 3
```

**lldp transmit**

Данной командой включается отправка LLDPDU на физическом интерфейсе.

Использование отрицательной формы команды (no), отправка LLDPDU на физическом интерфейсе отключается.

### Синтаксис

```
[no] lldp transmit
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отправка LLDP-пакетов выключена.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

### Пример

```
esr(config-if-gi)# lldp transmit
```

### show lldp neighbors

Данной командой выводится информация о подключенных устройствах, от которых получена информация по протоколу LLDP.

### Синтаксис

```
show lldp neighbors <IF>
```

### Параметры

<IF> – наименование физического интерфейса или группы физических интерфейсов, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

Допустимо использовать только физические интерфейсы (gigabitethernet или tengigabitethernet). Без указания интерфейса выводится информация о всех обнаруженных по LLDP устройствах.

### Необходимый уровень привилегий

10

## Командный режим

ROOT

### Пример

```

esr# show lldp neighbors
LLDP Neighbor Information:

Local Information:
  Index:          0
  Local Interface:  gi1/0/4

Neighbour Information:
  Chassis type:   mac
  Chassis ID:    a8:f9:4b:aa:8c:90
  Management ip: 192.168.1.5
  Management ip: fe80::2052:e5ff:fe36:226f
  Port type:     local
  Port ID:       gi1/0/8
  Port description: esr200-lldp-test
  Time to live:  120
  System name:   esr-200-test
  System Description: Eltex Router ESR-200 1.3.0 build 79 (date 14/08/2017 time 10:19:13)

System capabilities:
  Bridge:        false
  Router:        true
  Station:       true
  Wlan:          false

```

### show lldp statistics

Данной командой выводится статистика работы протокола LLDP на интерфейсах, на которых включен протокол LLDP.

### Синтаксис

```
show lldp statistics <IF>
```

### Параметры

<IF> – наименование интерфейса или группы интерфейсов, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#). Допустимо использовать только физические интерфейсы (gigabitethernet или tengigabitethernet). Без указания интерфейса выводится информация о всех обнаруженных по LLDP устройствах.

### Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
esr# show lldp statistics
Interface   Transmitted   Received   Discarded   Unrecognized   Ageout   Inserted   Deleted
-----
gi1/0/1     1             0          0           0             0        0          0
```

## Настройка зеркалирования

- [port monitor](#)
- [port monitor interface](#)
- [port monitor mode](#)
- [port monitor remote](#)
- [port monitor remote vlan](#)
- [show interfaces switch-port monitor](#)

### port monitor

Данной командой включается режим зеркалирования трафика.

Использование отрицательной формы команды (no) отключает режим зеркалирования трафика.

### Синтаксис

```
[no] port monitor
```

### Параметры

Отсутствуют.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

## Пример

```
esr(config-if-gi)# port monitor interface gigabitethernet 1/0/5
```

### port monitor interface

Данной командой определяются контролируемые порты.

Использование отрицательной формы команды (no) удаляет контролируемый порт.

## Синтаксис

```
port monitor interface <IF> <DIRECTION>
no port monitor
```

## Параметры

<IF> – физический интерфейс или группа физических интерфейсов, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<DIRECTION> – направление трафика:

- tx – зеркалирование только исходящего трафика;
- rx – зеркалирование только входящего трафика.

## Необходимый уровень привилегий

10

## Командный режим

```
CONFIG-IF-GI
CONFIG-IF-TE
CONFIG-IF-TWE
CONFIG-IF-FO
CONFIG-IF-HU
```

## Пример

```
esr(config-if-gi)# port monitor interface gigabitethernet 1/0/5
```

## port monitor mode

Данной командой определяется режим порта, передающего отзеркалированный трафик.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
port monitor mode <MODE>
no port monitor mode
```

## Параметры

<MODE> – режим:

- network – совмещенный режим передачи данных и зеркалирование;
- monitor-only – только зеркалирование.

## Значение по умолчанию

network



**Необходимый уровень привилегий**

10


**Командный режим**

CONFIG

**Пример**

```
esr(config)# port monitor mode monitor-only
```

**port monitor remote**

 В текущей версии ПО данный функционал поддерживается только на маршрутизаторах ESR-1000/1200/1500/1511/1700.

Данной командой включается режим удаленного зеркалирования (RSPAN).

Использование отрицательной формы команды (no) отключает удаленное зеркалирование (RSPAN).

**Синтаксис**

```
[no] port monitor remote
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

10

**Командный режим**


CONFIG-IF-GI

CONFIG-IF-TE

**Пример**

```
esr(config-if-gi)# port monitor remote
```

## port monitor remote vlan

 В текущей версии ПО данный функционал поддерживается только на маршрутизаторах ESR-1000/1200/1500/1511/1700.

Данной командой определяется VLAN, по которому будет передаваться отзеркаленный трафик. Использование отрицательной формы команды (no) удаляет указанный VLAN.

### Синтаксис

```
port monitor remote vlan <VID> <DIRECTION>  
no port monitor remote vlan <DIRECTION>
```

### Параметры

<VID> – идентификационный номер VLAN, задаётся в диапазоне [2...4094];

<DIRECTION> – направление трафика:

- tx – зеркалирование в указанный VLAN только исходящего трафика;
- rx – зеркалирование в указанный VLAN только входящего трафика.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# port monitor remote vlan 10
```

## show interfaces switch-port monitor

Команда используется для просмотра настроек зеркалирования.

### Синтаксис

```
show interfaces switch-port monitor
```

### Параметры

Отсутствуют.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
esr# show interfaces switch-port monitor
Port monitor mode:      network
RSPAN configuration RX: VLAN 222
RSPAN configuration TX: VLAN 222
Source Port      Destination Port  Type      RSPAN
-----
gi1/0/7          gi1/0/6          RX,TX     Enabled
```

## Настройка zabbix-agent и zabbix-проxy

- [active-server](#)
- [config-retrieve](#)
- [database](#)
- [enable](#)
- [hostname](#)
- [port](#)
- [remote-commands](#)
- [server](#)
- [server](#)
- [show zabbix-agent configuration](#)
- [show zabbix-proxy configuration](#)
- [source-address](#)
- [timeout](#)
- [zabbix-agent](#)
- [zabbix-proxy](#)

### active-server

Данной командой задается адрес и порт Zabbix-сервера для активных проверок.

Использование отрицательной формы команды (no) удаляет сервер для активных проверок.

### Синтаксис

```
active-server <ADDR> [ port <PORT> ]
```

```
no active-server
```

### Параметры

<ADDR> – IP-адрес сервера, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<PORT> – номер порта, указывается в диапазоне [1..65535].

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-ZABBIX-AGENT

### Пример

```
esr(config-zabbix-agent)# active-server 192.168.16.54
```

## config-retrieve

Данной командой задаётся интервал обновления данных конфигурации от сервера.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

### Синтаксис

```
config-retrieve <SEC>  
no config-retrieve
```

### Параметры

<SEC> – период времени в секундах, принимает значения [1..604800].

### Значение по умолчанию

60

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-ZABBIX-PROXY

### Пример

```
esr(config-zabbix-proxy)# config-retrieve 200
```

## database

Данной командой устанавливается место хранения базы данных zabbix-proxy.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
database <PATH>  
no database
```

### Параметры

<PATH> – место хранения базы данных zabbix-proxy.

### Значение по умолчанию

По умолчанию база данных zabbix хранится в энергозависимой памяти маршрутизатора.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-ZABBIX-PROXY

### Пример

```
esr(config-zabbix-agent)# database flash:data/zabbix.dat
```

### enable

Данной командой активируется функционал Zabbix-агента.

Использование отрицательной формы команды (no) отключает функционал Zabbix-агента.

### Синтаксис

[no] enable

### Параметры

Отсутствуют.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-ZABBIX-AGENT

CONFIG-ZABBIX-PROXY

### Пример

```
esr(config-zabbix-agent)# enable
```

### hostname

Данной командой задается имя агента. Hostname должен совпадать с именем хоста в панели управления Zabbix.

Использование отрицательной формы команды (no) устанавливает имя агента по умолчанию.

### Синтаксис

hostname <NAME>

no hostname

## Параметры

<NAME> – идентификатор имени системы, задается строкой до 63 символов.

## Значение по умолчанию

Совпадает с настроенным hostname маршрутизатора.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ZABBIX-AGENT

CONFIG-ZABBIX-PROXY

## Пример

```
esr(config-zabbix-agent)# hostname branch_12
```

## port

Данной командой задается TCP-порт, который будет использоваться для обращений Zabbix-сервера к агенту.

Использование отрицательной формы команды (no) удаляет значение TCP-порта.

## Синтаксис

```
port <PORT>
```

```
[no] port
```

## Параметры

<PORT> – номер порта, указывается в диапазоне [1024..32767].

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ZABBIX-AGENT

CONFIG-ZABBIX-PROXY

## Пример

```
esr(config-zabbix-agent)# port 20050
```

## remote-commands

Данной командой активируется возможность выполнение команд на маршрутизаторе с Zabbix-сервера.

Использование отрицательной формы команды (no) отключает возможность выполнение команд на маршрутизаторе с Zabbix-сервера.

## Синтаксис

```
[no] remote-commands
```

## Параметры

Отсутствуют.

## Значение по умолчанию

Отключено.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-ZABBIX-AGENT

CONFIG-ZABBIX-PROXY

## Пример

```
esr(config-zabbix-agent)# remote-commands
```

## server

Данной командой задается IP-адрес Zabbix-сервера для zabbix-agent, с которого разрешено принимать входящие соединения.

Использование отрицательной формы команды (no) удаляет IP-адрес Zabbix-сервера.

## Синтаксис

```
server <ADDR>
```

```
no server
```

## Параметры

<ADDR> – IP-адрес сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].



**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-ZABBIX-AGENT

**Пример**

```
esr(config-zabbix-agent)# server 192.168.18.54
```

**server**

Данной командой задается IP-адрес и порт Zabbix-сервера для zabbix-проху, с которого разрешено принимать входящие соединения.

Использование отрицательной формы команды (no) удаляет конфигурацию Zabbix-сервера для zabbix-проху.

**Синтаксис**

```
server <ADDR> [port <PORT>]
```

```
no server
```

**Параметры**

<ADDR> – IP-адрес сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<PORT> – номер UDP-порта, указывается в диапазоне [1024..32767].

**Значение по умолчанию**

<ADDR> – отсутствует;

<PORT> – 10051.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-ZABBIX-PROXY

**Пример**

```
esr(config-zabbix-agent)# server 192.168.18.54 port 1285
```

## show zabbix-agent configuration

Данной командой осуществляется вывод конфигурации Zabbix-агента.

### Синтаксис

```
show zabbix-agent configuration
```

### Параметры

Отсутствуют.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
esr# show zabbix-agent configuration
VRF:                --
State:               Enabled
Active server:       --
Active server port:  --
Hostname:            esr
Port:                --
Remote commands:     Disabled
Server:              192.168.18.54
Source address:      --
Timeout:             3
```

## show zabbix-proxy configuration

Данной командой осуществляется вывод конфигурации Zabbix-прокси.

### Синтаксис

```
show zabbix-proxy configuration
```

### Параметры

Отсутствуют.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

## Пример

```

esr# show zabbix-proxy configuration
VRF:                               --
State:                             Enabled
Hostname:                          ESR-10SLA
Port:                               --
Remote commands:                   Enabled
Server:                            192.168.32.228
Server port:                        --
Source address:                     --
Timeout:                            3
Configuration retrieve:             30

```

## source-address

Данной командой указывается адрес, с которого будет устанавливаться соединение с Zabbix-сервером. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
source-address <ADDR>
```

```
no source-address
```

## Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

## Значение по умолчанию

IP-адрес интерфейса, с которого отправляются пакеты на Zabbix-сервер.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ZABBIX-AGENT

CONFIG-ZABBIX-PROXY

## Пример

```

esr(config-zabbix-agent)# source-address 192.168.1.57

```

## timeout

Данной командой задается максимальное время на обработку запроса Zabbix-сервера.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
timeout  
no timeout
```

**Параметры**

<SEC> – период времени в секундах, принимает значения [1..30].

**Значение по умолчанию**

3

**Необходимый уровень привилегий**

10

**Командный режим**

```
CONFIG-ZABBIX-AGENT  
CONFIG-ZABBIX-PROXY
```

**Пример**

```
esr(config-zabbix-agent)# timeout 20
```

**zabbix-agent**

Данной командой осуществляется переход в режим конфигурирования Zabbix-агента в глобальном режиме или VRF.

Использование отрицательной формы команды (no) удаляет настройки Zabbix-агента в глобальном режиме или VRF.

**Синтаксис**

```
[no] zabbix-agent [ vrf <VRF> ]
```

**Параметры**

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа.

**Значение по умолчанию**

Отключен.

**Необходимый уровень привилегий**

10

**Командный режим**

```
CONFIG
```

## Пример

```
esr(config)# zabbix-agent
```

## zabbix-proxu

Данной командой осуществляется переход в режим конфигурирования Zabbix-proxu в глобальном режиме или VRF.

Использование отрицательной формы команды (no) удаляет настройки Zabbix-proxu в глобальном режиме или VRF.

## Синтаксис

```
[no] zabbix-proxy [ vrf <VRF> ]
```

## Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# zabbix-proxy
```

## 29 Настройка DHCP

- Управление DHCP-клиентом
  - ip address dhcp
  - ip dhcp client ignore
  - ip dhcp client lease-time
  - ip dhcp client reboot
  - ip dhcp client retry
  - ip dhcp client select-timeout
  - ip dhcp client timeout
  - ip dhcp client vendor-class-id
  - ip dhcp server address
  - renew dhcp
- Управление IPv6 DHCP-клиентом
  - ipv6 address dhcp
  - ipv6 dhcp client ignore
  - ipv6 dhcp client lease-time
  - ipv6 dhcp client reboot
  - ipv6 dhcp client retry
  - ipv6 dhcp client select-timeout
  - ipv6 dhcp client timeout
  - ipv6 dhcp client vendor-class-id
  - ipv6 dhcp server address
  - renew ipv6 dhcp
- Управление DHCP Relay агентом
  - ip dhcp information option
  - ip dhcp information option-insert
  - ip dhcp information option action
  - ip dhcp information option format-type access-node-id
  - ip dhcp information option format-type circuit-id
  - ip dhcp information option format-type option
  - ip dhcp information option format-type remote-id
  - ip dhcp information option suboption-type
  - ip dhcp-relay
  - ip helper-address
  - ip helper-address gateway-ip
  - ip helper-address vrrp-group
- Управление IPv6 DHCP Relay агентом
  - ipv6 dhcp-relay
  - ipv6 dhcp-relay interface
- Настройка и мониторинг DHCP-сервера
  - address
  - address-range
  - clear ip dhcp binding
  - default-lease-time
  - default-router
  - dns-server
  - domain-name
  - excluded-address-range
  - ip dhcp-server
  - ip dhcp-server dscp
  - ip dhcp-server pool
  - ip dhcp-server vendor-class-id
  - max-lease-time
  - netbios-name-server
  - network
  - next-server

- option
- show ip dhcp binding
- show ip dhcp server dscp
- show ip dhcp server pool
- show ip dhcp server vendor-specific
- suboption
- tftp-server
- vendor-specific
- vendor-specific-options
- Настройка и мониторинг IPv6 DHCP-сервера
  - address
  - address-range
  - default-lease-time
  - dns-server
  - domain-name
  - excluded-address-range
  - ipv6 dhcp-server
  - ipv6 dhcp-server pool
  - ipv6 dhcp-server vendor-class-id
  - max-lease-time
  - network
  - option
  - show ipv6 dhcp binding
  - show ipv6 dhcp server pool
  - show ipv6 dhcp server vendor-specific

## Управление DHCP-клиентом

### ip address dhcp

Данной командой включается получение динамического IP-адреса конфигурируемого интерфейса по протоколу DHCP.

Использование отрицательной формы команды (no) выключает получение динамического IP-адреса по протоколу DHCP.

### Синтаксис

```
[no] ip address dhcp
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Выключен.

### Необходимый уровень привилегий

10

**Командный режим**

CONFIG-IF-GI  
 CONFIG-IF-TE  
 CONFIG-IF-TWE  
 CONFIG-IF-FO  
 CONFIG-IF-HU  
 CONFIG-IF-OOB  
 CONFIG-IF-SUB  
 CONFIG-IF-QINQ  
 CONFIG-IF-PORT-CHANNEL  
 CONFIG-BRIDGE  
 CONFIG-GRE

**Пример**

```
esr(config-if-gi)# ip address dhcp
```

**ip dhcp client ignore**

Данной командой указываются DHCP-опции, которые будут игнорироваться клиентом.

**Синтаксис**

```
[no] ip dhcp client ignore <OPTION>
```

**Параметры**

<OPTION> – принимает следующие значения:

- dns-nameserver – DHCP-опция 23, список DNS-серверов;
- domain-name – DHCP-опция 24, имя домена;
- netbios-nameserver – DHCP-опция 44, список NetBios-серверов;
- router – DHCP-опция 3, список шлюзов по умолчанию;
- classless-static-route – DHCP-опция 121, список бесклассовых статических маршрутов;
- classful-static-route – DHCP-опция 33, список классовых статических маршрутов;
- tftp-server-address – DHCP-опция 66, имя TFTP-сервера (не применима для Bridge-интерфейсов);
- vendor-specific – DHCP-опция 17, информация определенная производителем.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI  
 CONFIG-IF-TE  
 CONFIG-IF-TWE



CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-OOB  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-GRE

### Пример

```
esr(config-if-gi)# ip dhcp client ignore router
```

### ip dhcp client lease-time

Данной командой устанавливается запрашиваемое время аренды сетевого адреса.  
Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ip dhcp client lease-time <TIME>  
no ip dhcp client lease-time
```

### Параметры

<TIME> – запрашиваемое время аренды, задаётся в виде DD:HH:MM, где:

- DD – дни, принимает значение [0..364];
- HH – часы, принимает значение [0..23];
- MM – минуты, принимает значение [0..59].

### Значение по умолчанию

1 день.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-OOB  
CONFIG-IF-SUB

CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-GRE

### Пример

```
esr(config-if-gi)# ip dhcp client lease-time 00:12:00
```

### ip dhcp client reboot

Данной командой задаётся время, в течение которого DHCP-клиент будет пытаться получить старый IP-адрес, перед тем как начать получать новый.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ip dhcp client reboot <SEC>  
no ip dhcp client reboot
```

### Параметры

<SEC> – период времени в секундах, принимает значение [1..600].

### Значение по умолчанию

10

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-OOB  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-GRE

## Пример

```
esr(config-if-gi)# ip dhcp client reboot 60
```

## ip dhcp client retry

Данной командой задаётся интервал, через который DHCP-клиент возобновит попытки получить IP-адрес, если было установлено, что DHCP-сервер не отвечает.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
ip dhcp client retry <SEC>
```

```
no ip dhcp client retry
```

## Параметры

<SEC> – период времени в секундах, принимает значение [1..600].

## Значение по умолчанию

300

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

## Пример

```
esr(config-if-gi)# ip dhcp client retry 180
```

## ip dhcp client select-timeout

Данной командой задаётся период времени, в течение которого DHCP-клиент будет выбирать среди предложений по аренде от серверов, если такие существуют. Это используется в сетях с несколькими DHCP-серверами. Клиенту в ответ на запрос IP-адреса может быть отправлено несколько предложений. Возможно, что одно из этих предложений предпочтительнее другого (например, одно предложение может иметь адрес, который клиент использовал ранее).

Клиент ждёт указанный период времени с момента отправки запроса на получение IP-адреса, на случай, если он получит несколько предложений от различных DHCP-серверов. По истечении указанного времени клиент принимает одно из предложений.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ip dhcp client select-timeout <SEC>
no ip dhcp client select-timeout
```

### Параметры

<SEC> – период времени в секундах, принимает значение [0..600].

### Значение по умолчанию

0 секунд – клиент примет первое полученное предложение.

### Необходимый уровень привилегий

10

### Командный режим

```
CONFIG-IF-GI
CONFIG-IF-TE
CONFIG-IF-TWE
CONFIG-IF-FO
CONFIG-IF-HU
CONFIG-IF-OOB
CONFIG-IF-SUB
CONFIG-IF-QINQ
CONFIG-IF-PORT-CHANNEL
CONFIG-BRIDGE
CONFIG-GRE
```

### Пример

```
esr(config-if-gi)# ip dhcp client select-timeout 30
```

## ip dhcp client timeout

Данной командой задаётся интервал, по истечении которого клиент считает, что DHCP-сервер недоступен. Если в базе данных IP-адресов клиента есть какие-либо арендованные адреса, срок аренды которых еще не истек, то клиент будет проверять последовательно каждый из них и, если найдет корректную, то IP-адрес из неё будет присвоен интерфейсу. Если нет действующих аренд в базе данных, то клиент будет повторно запрашивать IP-адрес по истечении интервала повтора (dhcp retry). Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ip dhcp client timeout <SEC>  
no ip dhcp client timeout
```

### Параметры

<SEC> – период времени в секундах, принимает значения [1 .. 600].

### Значение по умолчанию

60 секунд.

### Необходимый уровень привилегий

10

### Командный режим

```
CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-OOB  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-GRE
```

### Пример

```
esr(config-if-gi)# ip dhcp client timeout 300
```

## ip dhcp client vendor-class-id

Данной командой устанавливается значение DHCP Опции 60 для получения дополнительных настроек по DHCP Опции 43.

Использование отрицательной формы команды (no) отключает запрос данной опции.

### Синтаксис

```
ip dhcp client vendor-class-id <NAME>
```

```
no ip dhcp client vendor-class-id
```

### Параметры

<NAME> – идентификатор класса поставщика, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

### Пример

```
esr(config-if-gi)# ip dhcp client vendor-class-id ELTEX
```

### ip dhcp server address

Данной командой устанавливается IP-адрес DHCP-сервера, с которого будет получен IP-адрес (и другие опции). Предложения от других DHCP-серверов будут проигнорированы.

Использование отрицательной формы команды (no) удаляет установленный IP-адрес DHCP-сервера.

### Синтаксис

```
ip dhcp server address <ADDR>
```

```
no ip dhcp server address
```

### Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

**Пример**

```
esr(config-if-gi)# ip dhcp server address 10.10.10.1
```

**renew dhcp**

Данной командой отправляется запрос на обновление IP-адреса на по протоколу DHCP.

**Синтаксис**

```
renew dhcp <IF>
```

**Параметры**

<IF> – наименование интерфейса или группы интерфейсов, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

**Необходимый уровень привилегий**

10

**Командный режим**

ROOT

**Пример**

```
esr# renew dhcp gigabitethernet 1/0/1
```

## Управление IPv6 DHCP-клиентом

### ipv6 address dhcp

Данной командой включается получение динамического IPv6-адреса конфигурируемого интерфейса по протоколу IPv6 DHCP.

Использование отрицательной формы команды (no) выключает получение динамического IPv6-адреса по протоколу IPv6 DHCP.

### Синтаксис

```
[no] ipv6 address dhcp
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

### Пример

```
esr(config-if-gi)# ipv6 address dhcp
```

### ipv6 dhcp client ignore

Данной командой указываются DHCP-опции, которые будут игнорироваться клиентом.

### Синтаксис

```
ipv6 dhcp client ignore <OPTION>
```



```
no ipv6 dhcp client ignore
```

### Параметры

<OPTION> – принимает следующие значения:

- dns-nameserver – DHCP-опция 23, список DNS-серверов;
- domain-name – DHCP-опция 24, имя домена;
- vendor-specific – DHCP-опция 17, информация, определенная производителем.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

### Пример

```
esr(config-if-gi)# ipv6 dhcp client ignore dns-nameserver
```

### ipv6 dhcp client lease-time

Данной командой устанавливается запрашиваемое время аренды сетевого адреса.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ipv6 dhcp client lease-time <TIME>
```

```
no ipv6 dhcp client lease-time
```

### Параметры

<TIME> – запрашиваемое время аренды, задаётся в виде DD:HH:MM, где:

- DD – дни, принимает значение [0..364];
- HH – часы, принимает значение [0..23];
- MM – минуты, принимает значение [0..59].

### Значение по умолчанию

1 день.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

**Пример**

```
esr(config-if-gi)# ipv6 dhcp client lease-time 00:12:00
```

**ipv6 dhcp client reboot**

Данной командой задаётся время, в течение которого IPv6 DHCP-клиент будет пытаться получить старый IPv6-адрес перед тем, как начать получать новый.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
ipv6 dhcp client reboot <SEC>
```

```
no ipv6 dhcp client reboot
```

**Параметры**

<SEC> – период времени в секундах, принимает значение [1..600].

**Значение по умолчанию**

10

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE

### Пример

```
esr(config-if-gi)# ipv6 dhcp client reboot 60
```

### ipv6 dhcp client retry

Данной командой задаётся интервал, через который IPv6 DHCP-клиент возобновит попытки получить IPv6-адрес, если было установлено, что IPv6 DHCP-сервер не отвечает.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ipv6 dhcp client retry <SEC>  
no ipv6 dhcp client retry
```

### Параметры

<SEC> – период времени в секундах, принимает значение [1..600].

### Значение по умолчанию

300

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE

## Пример

```
esr(config-if-gi)# ipv6 dhcp client retry 180
```

### ipv6 dhcp client select-timeout

Данной командой задаётся период времени, в течение которого IPv6 DHCP-клиент будет выбирать среди предложений по аренде от серверов, если такие существуют. Это используется в сетях с несколькими IPv6 DHCP-серверами – в этом случае клиенту в ответ на запрос IPv6-адреса может быть отправлено несколько предложений. Возможно, что одно из этих предложений предпочтительнее другого (например, одно предложение может иметь адрес, который клиент использовал ранее).

Клиент ждёт указанный период времени с момента отправки запроса на получение IP-адреса, на случай, если он получит несколько предложений от различных DHCPv6-серверов. По истечении указанного времени клиент принимает одно из предложений.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ipv6 dhcp client select-timeout <SEC>
```

```
no ipv6 dhcp client select-timeout
```

### Параметры

<SEC> – период времени в секундах, принимает значение [0..600].

### Значение по умолчанию

0 – клиент примет первое предложенное предложение.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

## Пример

```
esr(config-if-gi)# ipv6 dhcp client select-timeout 30
```

## ipv6 dhcp client timeout

Данной командой задаётся интервал, по истечении которого клиент считает, что IPv6 DHCP-сервер недоступен. Если в базе данных IPv6-адресов клиента есть какие-либо арендованные адреса, срок аренды которых ещё не истек, то клиент будет проверять последовательно каждый из них, и если найдет корректную, то IPv6-адрес из неё будет присвоен интерфейсу. Если нет действующих аренд в базе данных, то клиент будет повторно запрашивать IPv6-адрес по истечении интервала повтора (dhcp retry).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
ipv6 dhcp client timeout <SEC>
```

```
no ipv6 dhcp client timeout
```

## Параметры

<SEC> – период времени в секундах, принимает значения [1..600].

## Значение по умолчанию

60 секунд.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

## Пример

```
esr(config-if-gi)# ipv6 dhcp client timeout 300
```

## ipv6 dhcp client vendor-class-id

Данной командой устанавливается значение DHCP опции 60 для получения дополнительных настроек по DHCP опции 43.

Использование отрицательной формы команды (no) отключает запрос данной опции.

### Синтаксис

```
ipv6 dhcp client vendor-class-id <NAME>  
no ipv6 dhcp client vendor-class-id
```

### Параметры

<NAME> – идентификатор класса поставщика, задаётся строкой до 31 символа.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

### Командный режим

```
CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE
```

### Пример

```
esr(config-if-gi)# ipv6 dhcp client vendor-class-id ELTEX
```

## ipv6 dhcp server address

Данной командой устанавливается IPv6-адрес DHCP-сервера, с которого будет получен IPv6-адрес (и другие опции). Предложения от других DHCP-серверов будут проигнорированы.

Использование отрицательной формы команды (no) удаляет установленный IPv6-адрес DHCP-сервера.

### Синтаксис

```
ipv6 dhcp server address <IPV6-ADDR>
```

```
no ipv6 dhcp server address
```

### Параметры

<IPv6-ADDR> – IPv6-адрес, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

### Пример

```
esr(config-if-gi)# ipv6 dhcp server address fc00::1
```

### renew ipv6 dhcp

Данной командой отправляется запрос на обновление IPv6-адреса по протоколу DHCP.

### Синтаксис

```
renew ipv6 dhcp <IF>
```

### Параметры

<IF> – наименование интерфейса или группы интерфейсов, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

### Необходимый уровень привилегий

10

### Командный режим

ROOT

## Пример

```
esr# renew ipv6 dhcp gigabitethernet 1/0/1
```

## Управление DHCP Relay агентом

### ip dhcp information option

Данной командой включается обработка опции 82 DHCP Relay агентом.

Использование отрицательной формы команды (no) отключает обработку опции 82 DHCP Relay агентом.

### Синтаксис

```
[no] ip dhcp information option
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

## Пример

```
esr(config)# ip dhcp information option
```

### ip dhcp information option-insert

Данной командой включается вставка DHCP Relay агентом, работающим на multipoint GRE туннеле ESR в роли NHRP NHS, 82 опции в DHCP-запросы от NHRP NHS. В добавляемой опции указывается NBMA-адрес NHRP NHS.

Использование отрицательной формы команды (no) отключает вставка опции 82 DHCP Relay агентом.

### Синтаксис

```
[no] ip dhcp information option-insert
```

### Параметры

Команда не содержит параметров.



**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-GRE

**Пример**

```
esr(config-gre)# ip dhcp information option-insert
```

**ip dhcp information option action**

Данная команда позволяет изменить решение, которое будет принято маршрутизатором при получении пакета с установленной опцией 82.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
ip dhcp information option action { replace | keep | drop }  
no ip dhcp information option action
```

**Параметры**

- replace – заменить установленную опцию 82 новой;
- keep – оставить установленную опцию 82 без изменений;
- drop – удалить пакет, если в нем обнаружена установленная опция 82.

**Значение по умолчанию**

keep

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

### Пример

```
esr(config-if-gi)# ip dhcp information option action drop
```

### ip dhcp information option format-type access-node-id

Данная команда позволяет задать Access-node-ID в поле Circuit ID опции 82 при использовании формата tr101.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
[no] ip dhcp information option format-type access-node-id <NODE-ID>
```

### Параметры

<NODE-ID> – значение Access-node-ID, задаётся строкой до 31 символа.

### Значение по умолчанию

Hostname устройства.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# ip dhcp information option format-type access-node-id R1
```

### ip dhcp information option format-type circuit-id

Данная команда позволяет задать значение поля Circuit ID опции 82 при использовании любого формата.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
[no] ip dhcp information option format-type circuit-id <CIRCUIT-ID>
```

### Параметры

<CIRCUIT-ID> – значение Circuit ID, задаётся строкой до 63 символов.

**Значение по умолчанию**

Определяется используемым форматом.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

**Пример**

```
esr(config-if-gi)# ip dhcp information option format-type circuit-id gi1_0_1
```

**ip dhcp information option format-type option**

Данная команда позволяет задать формат идентификатора порта в поле Circuit ID опции 82 при использовании формата tr101.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
[no] ip dhcp information option format-type option <FORMAT> [ delimiter <DELIMITER> ]
```

**Параметры**

<FORMAT> – формат идентификатора порта, принимает одно из следующих значений:

- sp – строка содержащая номер слота и порта;
- sv – строка содержащая номер слота и VLAN;
- pv – строка содержащая номер порта и VLAN;
- spv – строка содержащая номер слота, порта и VLAN;
- bin – бинарный формат: VLAN(4 байта), слот(1 байт), порт(4 байта).

<DELIMITER> – разделитель между параметрами в строке, может принимать одно из следующих значений [ . ; # / space ].

**Значение по умолчанию**

Формат – spv

Разделитель «:» для форматов: sp, sv, pv

Разделитель «/» и «:» для формата spv

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# ip dhcp information option format-type option spv delimiter #
```

### ip dhcp information option format-type remote-id

Данная команда позволяет задать значение поля Remote ID опции 82 при использовании любого формата.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
[no] ip dhcp information option format-type remote-id <REMOTE-ID>
```

### Параметры

<REMOTE-ID> – значение Remote ID, задаётся строкой до 63 символов.

### Значение по умолчанию

Определяется используемым форматом.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

## Пример

```
esr(config-if-gi)# ip dhcp information option format-type remote-id R1
```

## ip dhcp information option suboption-type

Данной командой выбирается формат опции 82.

Использование отрицательной формы команды (no) устанавливает формат по умолчанию.

## Синтаксис

```
[no] ip dhcp information option suboption-type { tr101 | custom }
```

## Параметры

tr101 – формат, рекомендованный TR-101 опции 82 согласно синтаксису, принятому в рекомендациях TR-101, приведен в [таблице 10](#);

custom – формат опции 82 в соответствии с форматом, приведенном в [таблице 11](#).

Таблица 10 – Формат полей опции 82 согласно рекомендациям TR-101

Поле	Передаваемая информация
Circuit ID	<p>Access-Node-Id, по умолчанию hostname устройства (формат можно изменить командой «ip dhcp information option format-type access-node-id», описанной в разделе <a href="#">ip dhcp information option format-type access-node-id</a>)</p> <p>Строка вида eth &lt;stacked/slotid/interfaceid&gt;:&lt;vlan&gt; (формат можно изменить командой «ip dhcp information option format-type option», описанной в разделе <a href="#">ip dhcp information option format-type option</a>)</p> <p>Последний байт – номер порта, к которому подключено устройство, отправляющее dhcp-запрос.</p>
Remote agent ID	<p>Enterprise number – 0089c1</p> <p>MAC-адрес устройства</p>

Поле	Передаваемая информация
Circuit ID	Длина (1 байт) Тип Circuit ID Длина (1 байт) VLAN (2 байта) Номер модуля (1 байт) Номер порта (4 байта)
Remote agent ID	Длина (1 байт) Тип Remote ID (1 байт) Длина (1 байт) MAC-адрес коммутатора

**Значение по умолчанию**

tr101

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip dhcp information option suboption-type custom
```

**ip dhcp-relay**

Данная команда включает агент DHCP-relay.

Использование отрицательной формы команды (no) отключает агент DHCP-relay.

**Синтаксис**

```
[no] ip dhcp-relay
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# ip dhcp-relay
```

## ip helper-address

Данной командой указывается IP-адрес DHCP-сервера, которому будут отправляться DHCP Discover пакеты, перехваченные DHCP Relay-агентом.


Использование отрицательной формы команды (no) удаляет IP-адрес из списка DHCP-серверов для DHCP Relay-агента.

## Синтаксис

```
[no] ip helper-address <IP>
```

## Параметры

<IP> – IP-адрес DHCP-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Можно указать до 6 IP-адресов, список задаётся через запятую.

 Для GRE можно указать до 4.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

## Пример

```
esr(config-if-gi)# ip helper-address 10.10.10.1
```

### ip helper-address gateway-ip

Данной командой выставляется необходимый IP-адрес в поле Gateway-IP в DHCP Offer пакетах, отправляемых DHCP Relay-агентом.

Использование отрицательной формы команды (no) выставляет ip-address в поле Gateway-IP по умолчанию.

### Синтаксис

```
ip helper-address gateway-ip <ADDR>
```

```
no ip helper-address gateway-ip
```

### Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

## Пример

```
esr(config-if-gi)# ip helper-address gateway-ip 10.10.10.1
```

### ip helper-address vrrp-group

Данной командой запрещается отправка DHCP Discover пакетов, перехваченных DHCP Relay-агентом, если VRRP-группа находится в состоянии DOWN.

Использование отрицательной формы команды (no) разрешает отставку DHCP Discover пакетов, перехваченные DHCP Relay-агентом, если VRRP-группа находится в состоянии DOWN.



**Синтаксис**

```
ip helper-address vrrp-group <GRID>
no ip helper-address vrrp-group
```

**Параметры**

<GRID> – идентификатор группы VRRP-маршрутизатора, принимает значения [1..32].

**Необходимый уровень привилегий**

10

**Командный режим**

```
CONFIG-IF-GI
CONFIG-IF-TE
CONFIG-IF-TWE
CONFIG-IF-FO
CONFIG-IF-HU
CONFIG-IF-SUB
CONFIG-IF-QINQ
CONFIG-IF-PORT-CHANNEL
CONFIG-BRIDGE
```

**Пример**

```
esr(config-if-gi)# ip helper-address vrrp-group 7
```

Таблица 11 – Формат полей опции 82 режима custom

**Управление IPv6 DHCP Relay агентом****ipv6 dhcp-relay**

Данная команда включает IPv6 DHCP-агент.

Использование отрицательной формы команды (no) выключает IPv6 DHCP-агент.

**Синтаксис**

```
[no] ipv6 dhcp-relay
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ipv6 dhcp-relay
```

**ipv6 dhcp-relay interface**

Данной командой указывается исходящий интерфейс и IPv6-адрес DHCP-сервера, которому будут отправляться DHCP Solicit-пакеты, перехваченные IPv6 DHCP Relay-агентом.

Использование отрицательной формы команды (no) удаляет исходящий интерфейс из списка интерфейсов для IPv6 DHCP Relay-агента.

**Синтаксис**

```
ipv6 dhcp-relay interface <IF> [ <IPV6-ADDR> ]
```

```
no ipv6 dhcp-relay interface <IF>
```

**Параметры**

<IF> – наименование интерфейса или группы интерфейсов, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

<IPV6-ADDR> – IPv6-адрес DHCP-сервера, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF]. Если адрес не задан, то пакеты отправляются на ff02::1:2. Все IPv6 DHCP-сервера и агенты в локальном сетевом сегменте.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

## Пример

```
esr(config-if-gi)# ipv6 dhcp-relay interface gigabitethernet 1/0/2 fc00::1
```

## Настройка и мониторинг DHCP-сервера

### address

Данная команда позволяет добавить IP-адрес для определенного физического адреса к пулу адресов конфигурируемого DHCP-сервера.

Использование отрицательной формы команды (no) удаляет указанный IP-адрес.

### Синтаксис

```
address <ADDR> { mac-address <MAC> | client-identifier <CLIENT-ID> } [DESCRIPTION]
no address <ADDR>
```

### Параметры

<ADDR> – IP-адрес клиента, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Если использовать команду для удаления, то при указании значения «all» будут удалены все IP-адреса;

<MAC> – MAC-адрес клиента, которому будет выдан IP-адрес, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF];

<CLIENT-ID> – идентификатор клиента согласно DHCP option 61;

<DESCRIPTION> – описание заданного IP-адреса, задаётся строкой до 255 символов.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-DHCP-SERVER

## Пример

```
esr(config-dhcp-server)# address 192.168.3.21 mac-address A8:F9:4B:AA:00:40
```

## address-range

Данная команда позволяет добавить диапазон IP-адресов к пулу адресов, конфигурируемого DHCP-сервера.

Использование отрицательной формы команды (no) удаляет указанный диапазон IP-адресов.

### Синтаксис

```
address-range <FROM-ADDR>-<TO-ADDR>
no address-range { <FROM-ADDR>-<TO-ADDR> | all }
```

### Параметры

<FROM-ADDR> – начальный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<TO-ADDR> – конечный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Можно указать до 32 диапазонов IP-адресов, список задаётся через запятую.

all – удалить все сконфигурированные диапазоны IP-адресов.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-DHCP-SERVER

### Пример

```
esr(config-dhcp-server)# address-range 192.168.3.1-192.168.3.20,192.168.3.24
```

## clear ip dhcp binding

Данная команда позволяет очистить таблицу, выданную DHCP-сервером IP-адресов.

### Синтаксис

```
clear ip dhcp binding [ <ADDR> ] [vrf <VRF>]
```

### Параметры

<ADDR> – IP-адрес, опциональный параметр, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Если в команде задан параметр <ADDR>, то будет отображена информация связанная только с указанным адресом.

<VRF> – имя экземпляра VRF, по которому будет выводиться информация. Задаётся строкой до 31 символа.

## Необходимый уровень привилегий

10

## Командный режим

ROOT

## Пример

```
esr# clear ip dhcp binding
```

## default-lease-time

Данная команда позволяет задать время аренды, на которое клиенту будет выдан IP-адрес, если клиент не запрашивал определенное время аренды. Использование отрицательной формы команды устанавливает значение по умолчанию.

## Синтаксис

```
default-lease-time <TIME>
```

```
no default-lease-time
```

## Параметры

<TIME> – время аренды IP-адреса, в формате DD:HH:MM, где:

- DD – количество дней, принимает значения [0..364];
- HH – количество часов, принимает значения [0..23];
- MM – количество минут, принимает значения [0..59].

## Значение по умолчанию

12 часов

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-DHCP-SERVER

## Пример

```
esr(config-dhcp-server)# default-lease-time 00:04:00
```

## default-router

Данная команда позволяет задать список IP-адресов шлюзов по умолчанию, которые DHCP-сервер будет сообщать клиентам, используя DHCP опцию 3.

Использование отрицательной формы команды (no) удаляет указанные адреса из списка шлюзов.

### Синтаксис

```
default-router <ADDR>  
no default-router { <ADDR> | all }
```

### Параметры

<ADDR> – IP-адрес шлюза по умолчанию, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Можно указать до 8 IP-адресов, список задаётся через запятую. Если использовать команду для удаления, то при указании значения «all» будут удалены все шлюзы по умолчанию.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-DHCP-SERVER

### Пример

```
esr(config-dhcp-server)# default-router 192.168.3.1,192.168.3.2
```

### dns-server

Данная команда позволяет задать список IP-адресов DNS-серверов. Список передаётся клиентам в составе DHCP-опций.

Использование отрицательной формы команды удаляет указанный DNS-сервер из списка.

### Синтаксис

```
dns-server <ADDR>  
no dns-server { <ADDR> | all }
```

### Параметры

<ADDR> – IP-адрес DNS-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Можно указать до 8 IP-адресов, список задаётся через запятую. Если использовать команду для удаления, то при указании значения «all» будут удалены все DNS-серверы.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-DHCP-SERVER

## Пример

```
esr(config-dhcp-server)# dns-server 8.8.8.8,8.8.4.4
```

## domain-name

Данная команда позволяет задать DNS-имя сетевого домена. Имя домена передается клиентам в составе DHCP-опций 15.

Использование отрицательной формы команды (no) удаляет установленное имя домена.

## Синтаксис

```
domain-name <NAME>
```

```
no domain-name
```

## Параметры

<NAME> – DNS-имя домена клиента, задаётся строкой до 31 символа.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-DHCP-SERVER

## Пример

```
esr(config-dhcp-server)# domain-name eltex.loc
```

## excluded-address-range

Данная команда позволяет исключить диапазон IP-адресов из пула адресов, конфигурируемого DHCP-сервера.

Использование отрицательной формы команды (no) удаляет указанный диапазон IP-адресов.

## Синтаксис

```
[no] excluded-address-range <FROM-ADDR>-<TO-ADDR>
```

## Параметры

<FROM-ADDR> – начальный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

<TO-ADDR> – конечный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Можно указать до 32 диапазонов IP-адресов, список задаётся через запятую.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-DHCP-SERVER

**Пример**

```
esr(config-dhcp-server)# excluded-address-range 192.168.3.1-192.168.3.20,192.168.3.24
```

**ip dhcp-server**

Данная команда включает DHCP-сервер.

Использование отрицательной формы команды (no) выключает DHCP-сервер.

**Синтаксис**

```
[no] ip dhcp-server [vrf <VRF>]
```

**Параметры**

<VRF> – имя экземпляра VRF, в рамках которого будет работать DHCP-сервер. Задается строкой до 31 символа.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip dhcp-server
```

**ip dhcp-server dscp**

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов DHCP-сервера.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.



**Синтаксис**

```
ip dhcp-server dscp <DSCP>
no ip dhcp-server dscp
```

**Параметры**

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

**Значение по умолчанию**

61

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip dhcp-server dscp 40
```

**ip dhcp-server pool**

Команда используется для создания пула IP-адресов DHCP-сервера и перехода в режим его конфигурирования.

Использование отрицательной формы команды (no) удаляет указанный пул IP-адресов.

**Синтаксис**

```
ip dhcp-server pool <NAME> [vrf <VRF>]
no ip dhcp-server pool { <NAME> | all }
```

**Параметры**

<NAME> – имя пула IP-адресов DHCP-сервера, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все пулы IP-адресов. Если использовать команду для удаления, то при указании значения «all» будут удалены все IP-адреса;

<VRF> – имя экземпляра VRF, в рамках которого будет работать данный пул IP-адресов DHCP-сервера. Задаётся строкой до 31 символа.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

## Пример

```
esr(config)# ip dhcp-server pool lan
```

## ip dhcp-server vendor-class-id

Данная команда необходима для создания идентификатора класса поставщика (DHCP опция 60) и перехода в режим его конфигурирования.

Использование отрицательной формы команды (no) удаляет указанный идентификатор класса поставщика.

## Синтаксис

```
[no] ip dhcp-server vendor-class-id <NAME>
```

## Параметры

<NAME> – идентификатор класса поставщика, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все идентификаторы класса поставщика.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# ip dhcp-server vendor-class-id ELTEX
```

## max-lease-time

Данная команда позволяет задать максимальное время аренды IP-адресов. Если DHCP-клиент запрашивает время аренды, превосходящее максимальное значение, то будет установлено время, заданное этой командой. Использование отрицательной формы команды устанавливает значение по умолчанию.

## Синтаксис

```
max-lease-time <TIME>
```

```
no max-lease-time
```

## Параметры

<TIME> – максимальное время аренды IP-адреса, задаётся в формате DD:HH:MM, где:

- DD – количество дней, принимает значения [0..364];
- HH – количество часов, принимает значения [0..23];
- MM – количество минут, принимает значения [0..59].

## Значение по умолчанию

1 день.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-DHCP-SERVER

## Пример

```
esr(config-dhcp-server)# max-lease-time 00:16:00
```

## netbios-name-server

Данная команда позволяет сконфигурировать 44 опцию DHCP (задает IP-адрес NetBIOS-сервера).

Использование отрицательной формы команды (no) выключает передачу IP-адреса NetBIOS-сервера (44 опцию).

## Синтаксис

```
[no] netbios-name-server <ADDR>
```

## Параметры

<ADDR> – IP-адрес NetBIOS-сервера задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Можно задать до 4 IP-адресов. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все IP-адреса.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-DHCP-SERVER

## Пример

```
esr(config-dhcp-server)# netbios-name-server 192.168.45.15
```

## network

Данная команда задает IP-адрес и маску для подсети, из которой будет выделен пул IP-адресов. Использование отрицательной формы команды (no) удаляет настройки подсети в пуле.

### Синтаксис

```
network <ADDR/LEN>  
no network
```

### Параметры

<ADDR/LEN> – IP-подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-DHCP-SERVER

### Пример

```
esr(config-dhcp-server)# network 192.168.3.0/24
```

## next-server

Данная команда позволяет сконфигурировать 66 опцию DHCP.

Использование отрицательной формы команды (no) выключает передачу соответствующей опции.

### Синтаксис

```
next-server <ADDR>  
no next-server
```

### Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Можно задать только один адрес.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-DHCP-SERVER

## Пример

```
esr(config-dhcp-server)# next-server 172.16.32.1
```

## option

Данная команда позволяет задать опции DHCP.

Использование отрицательной формы удаляет заданную опцию.

## Синтаксис

```
option <CODE> { ip-address <ADDR> | ascii-text <STRING> | hex-bytes <HEX> }
no option { <CODE> | all }
```

## Параметры

<CODE> – код опции DHCP, задается в форме числа [0..255].

<HEX> – специфическая информация поставщика, задается в шестнадцатеричном формате до 255 байт.

<ADDR> – IP-адрес, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Можно указать до 15 IP-адресов, список задается через запятую.

<STRING> – текст, задается строкой от 1 до 255 символов.

all – удаляет все опции.

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-DHCP-SERVER

## Пример

```
esr(config-dhcp-server)# option 43 ip-address 10.10.1.1
```

## show ip dhcp binding

Данная команда позволяет посмотреть выданные DHCP-сервером IP-адреса. В примере ниже во второй колонке отображается значение опции 61 DHCP, указанной DHCP-клиентом, а в случае её отсутствия указывается MAC-адрес.

## Синтаксис

```
show ip dhcp binding [ <ADDR> ] [vrf <VRF>]
```

## Параметры

<ADDR> – IP-адрес, опциональный параметр, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Если в команде задан параметр <ADDR>, то будет отображена информация связанная только с указанным адресом.

<VRF> – имя экземпляра VRF, по которому будет выводиться информация. Задаётся строкой до 31 символа.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```

esr# show ip dhcp binding
IP address          MAC / Client ID                                     Binding type
Lease expires at
-----
-----
172.16.1.11         0x00656c7465782d636339642e613237312e373138342d6772655f31  active
2024-04-04 12:02:25
172.16.1.12         0x00656c7465782d613866392e346261622e643730302d6772655f31  active
2024-04-04 12:02:27
192.168.1.2         cc:9d:a2:71:71:c9                                       active
2024-04-04 17:04:27
192.168.1.3         a8:f9:4b:ab:d7:01                                       active
2024-04-04 17:04:30

```

## show ip dhcp server dscp

Данная команда позволяет посмотреть значение DSCP для сообщений DHCP-сервера.

## Синтаксис

```
show ip dhcp server dscp
```

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
esr # show ip dhcp server dscp
DSCP:    32
```

## show ip dhcp server pool

Данная команда позволяет посмотреть настроенные пулы IP-адресов. При указании имени выводится информация только для заданного пула.

### Синтаксис

```
show ip dhcp server pool [ <POOL_NAME> ]
```

### Параметры

<POOL\_NAME> – имя пула, опциональный параметр.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

## Пример

```
esr# show ip dhcp server pool lan-pool
name:                lan-pool
network:              192.168.1.0/24
address-ranges:      192.168.1.2-192.168.1.254
default-router:      192.168.1.1
max lease time (d:h:m): 001:0:0 (day:hour:min)
default lease time (d:h:m): 000:12:0 (day:hour:min)
```

## show ip dhcp server vendor-specific

Данная команда позволяет посмотреть настроенные DHCP опции 43 и 60.

### Синтаксис

```
show ip dhcp server vendor-specific
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

1

## Командный режим

ROOT

### Пример

```

esr# show ip dhcp server vendor-specific
Vendor ID      Vendor options
-----
ELTEX          0x0b0931302e312e39302e32

```

### suboption

Данная команда позволяет задать подопцию опции 43.

Использование отрицательной формы команды (no) удаляет заданную подопцию.

### Синтаксис

```

suboption <CODE> {ascii-text <TEXT> | hex-bytes <HEX>}
no suboption <CODE> | all

```

### Параметры

<CODE> – код подопции, задается в форме числа [0..255];

<STRING> – текст, задаётся строкой от 1 до 255 символов;

<HEX> – специфическая информация поставщика, задаётся в шестнадцатеричном формате до 255 байт;

all – удаляет все опции.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-DHCP-SERVER-VENDOR-SPECIFIC

### Пример

```

esr(config-dhcp-server-vendor-specific)# suboption 15 ascii-text "http://192.168.1.1:8042"

```

### tftp-server

Данная команда позволяет задать IP-адрес TFTP-сервера. Адрес передаётся клиентам в составе DHCP-опции 150.

Использование отрицательной формы команды (no) удаляет установленный IP-адрес TFTP-сервера.



**Синтаксис**

```
tftp-server <ADDR>  
no tftp-server
```

**Параметры**

<ADDR> – IP-адрес TFTP-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

**Необходимый уровень привилегий**

10

**Командный режим**

```
config-dhcp-server
```

**Пример**

```
esr(config-ipv6-dhcp-server)# tftp-server 192.168.0.1
```

**vendor-specific**

Данная команда позволяет задать специфическую информацию поставщика (DHCP опция 43).

Использование отрицательной формы команды (no) удаляет специфическую информацию поставщика.

**Синтаксис**

```
vendor-specific  
no vendor-specific
```

**Параметры**

Отсутствуют.

**Необходимый уровень привилегий**

10

**Командный режим**

```
CONFIG-DHCP-SERVER
```

**Пример**

```
esr(config-dhcp-server)# vendor-specific  
esr(config-dhcp-server-vendor-specific)#
```

## vendor-specific-options

Данная команда позволяет задать специфическую информацию поставщика (DHCP опция 43).

Использование отрицательной формы команды (no) удаляет специфическую информацию поставщика.

### Синтаксис

```
vendor-specific-options <HEX>
no vendor-specific-options
```

### Параметры

<HEX> – специфическая информация поставщика, задаётся в шестнадцатеричном формате до 128 СИМВОЛОВ.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-DHCP-VENDOR-ID

### Пример

```
esr(config-dhcp-vendor-id)# vendor-specific-options 0b0931302e312e39302e320
```

## Настройка и мониторинг IPv6 DHCP-сервера

### address

Данная команда позволяет добавить IPv6-адрес для определенного физического адреса к пулу адресов конфигурируемого DHCP-сервера.

Использование отрицательной формы команды (no) удаляет указанный IPv6-адрес.

### Синтаксис

```
address <IPV6-ADDR> mac-address <MAC>
no address <IPV6-ADDR>
```

### Параметры

<IPV6-ADDR> – IPv6-адрес клиента, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF]. Если использовать команду для удаления, то при указании значения «all» будут удалены все IPv6-адреса;

<MAC> – MAC-адрес клиента, которому будет выдан IPv6-адрес, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF].

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IPV6-DHCP-SERVER

**Пример**

```
esr(config-ipv6-dhcp-server)# address fc00::2 mac-address A8:F9:4B:AA:00:40
```

**address-range**

Данная команда позволяет добавить диапазон IPv6-адресов к пулу адресов конфигурируемого DHCP-сервера.

Использование отрицательной формы команды (no) удаляет указанный диапазон IPv6-адресов.

**Синтаксис**

```
[no] address-range <FROM-ADDR>-<TO-ADDR>
```

```
no address-range all
```

**Параметры**

<FROM-ADDR> – начальный IPv6-адрес диапазона, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<TO-ADDR> – конечный IP-адрес диапазона, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

Можно указать до 32 диапазонов IPv6-адресов, список задаётся через запятую.

all – удалить все сконфигурированные диапазоны IPv6-адресов.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IPV6-DHCP-SERVER

**Пример**

```
esr(config-ipv6-dhcp-server)# address-range fc00::1-fc00:12,fc00::15-fc00::25
```

**default-lease-time**

Данная команда позволяет задать время аренды, на которое клиенту будет выдан IPv6-адрес, если клиент не запрашивал определенное время аренды.

Использование отрицательной формы команды устанавливает значение по умолчанию.

## Синтаксис

```
default-lease-time <TIME>
no default-lease-time
```

## Параметры

<TIME> – время аренды IP-адреса, в формате DD:HH:MM, где:

- DD – количество дней, принимает значения [0..364];
- HH – количество часов, принимает значения [0..23];
- MM – количество минут, принимает значения [0..59].

## Значение по умолчанию

12 часов.

## Необходимый уровень привилегий

1

## Командный режим

CONFIG-IPV6-DHCP-SERVER

## Пример

```
esr(config-ipv6-dhcp-server)# default-lease-time 00:04:00
```

## dns-server

Данная команда позволяет задать список IPv6-адресов DNS-серверов. Список передаётся клиентам в составе DHCP-опций.

Использование отрицательной формы команды удаляет указанный DNS-сервер из списка.

## Синтаксис

```
dns-server <IPV6-ADDR>
no dns-server { <IPV6-ADDR> | all }
```

## Параметры

<IPV6-ADDR> – IPv6-адрес DNS-сервера, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF]. Можно указать до 8 IPv6-адресов, список задаётся через запятую. Если использовать команду для удаления, то при указании значения «all» будут удалены все DNS-серверы.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IPV6-DHCP-SERVER

### Пример

```
esr(config-ipv6-dhcp-server)# dns-server 8.8.8.8,8.8.4.4
```

### domain-name

Данная команда позволяет задать имя сетевого домена. Имя домена передаётся клиентам в составе DHCP-опций.

Использование отрицательной формы команды (no) удаляет установленное имя домена.

### Синтаксис

domain-name <NAME>

no domain-name

### Параметры

<NAME> – имя домена клиента, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-IPV6-DHCP-SERVER

### Пример

```
esr(config-ipv6-dhcp-server)# domain-name eltex.loc
```

### excluded-address-range

Данная команда позволяет исключить диапазон IPv6-адресов из пула адресов, конфигурируемого DHCP-сервера.

Использование отрицательной формы команды (no) удаляет указанный диапазон IPv6-адресов.

### Синтаксис

[no] excluded-address-range <FROM-ADDR>-<TO-ADDR>

### Параметры

<FROM-ADDR> – начальный IPv6-адрес диапазона, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<TO-ADDR> – конечный IPv6-адрес диапазона, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

Можно указать до 32 диапазонов IPv6-адресов, список задаётся через запятую.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IPV6-DHCP-SERVER

### Пример

```
esr(config-ipv6-dhcp-server)# excluded-address-range 10::1-10::9
```

### ipv6 dhcp-server

Данная команда включает IPv6 DHCP-сервер.

Использование отрицательной формы команды (no) выключает IPv6 DHCP-сервер.

### Синтаксис

```
[no] ipv6 dhcp-server [ vrf <VRF> ]
```

### Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать DHCP-сервер.

### Значение по умолчанию

Выключен.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# ipv6 dhcp-server
```

### ipv6 dhcp-server pool

Команда используется для создания пула IPv6-адресов DHCP-сервера и перехода в режим его конфигурирования.

Использование отрицательной формы команды (no) удаляет указанный пул IPv6-адресов.

### Синтаксис

```
[no] ipv6 dhcp-server pool <NAME> [vrf <VRF>]
```

### Параметры

<NAME> – имя пула IPv6-адресов DHCP-сервера, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все пулы IPv6-адресов;

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать данный пул IP-адресов DHCP-сервера.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# ipv6 dhcp-server pool lan
```

### ipv6 dhcp-server vendor-class-id

Данная команда необходима для создания идентификатора класса поставщика (DHCP опция 60) и перехода в режим его конфигурирования.

Использование отрицательной формы команды (no) удаляет указанный идентификатор класса поставщика.

### Синтаксис

```
[no] ipv6 dhcp-server vendor-class-id <NAME>
```

### Параметры

<NAME> – идентификатор класса поставщика, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все идентификаторы класса поставщика.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

## Пример

```
esr(config)# ipv6 dhcp-server vendor-class-id ELTEX
```

## max-lease-time

Данная команда позволяет задать максимальное время аренды IPv6-адресов. Если IPv6 DHCP-клиент запрашивает время аренды, превосходящее максимальное значение, то будет установлено время, заданное этой командой. Использование отрицательной формы команды устанавливает значение по умолчанию.

## Синтаксис

```
max-lease-time <TIME>
```

```
no max-lease-time
```

## Параметры

<TIME> – максимальное время аренды IP-адреса, задаётся в формате DD:HH:MM, где:

- DD – количество дней, принимает значения [0..364];
- HH – количество часов, принимает значения [0..23];
- MM – количество минут, принимает значения [0..59].

## Значение по умолчанию

1 день.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IPV6-DHCP-SERVER

## Пример

```
esr(config-ipv6-dhcp-server)# max-lease-time 00:16:00
```

## network

Данная команда задает IPv6-адрес и маску для подсети, из которой будет выделен пул IP-адресов. Использование отрицательной формы команды (no) удаляет настройки подсети в пуле.

## Синтаксис

```
network <IPV6-ADDR/LEN>
```

```
no network
```



## Параметры

<IPV6-ADDR/LEN> – IP-адрес и префикс подсети, задаётся в виде X:X:X::X/EE, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IPV6-DHCP-SERVER

## Пример

```
esr(config-ipv6-dhcp-server)# network fc00::/110
```

## option

Данная команда позволяет задать опции DHCP.

Использование отрицательной формы удаляет данную опцию.

## Синтаксис

```
option <CODE> { ip-address <IPV6-ADDR> | ascii-text <STRING> | hex-bytes <HEX> }
no option { <CODE> | all }
```

## Параметры

<CODE> – код опции DHCP, задается в форме числа [1..255];

<HEX> – специфическая информация поставщика, задаётся в шестнадцатеричном формате до 255 символов;

<IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF]. Можно указать до 8 IP-адресов, список задаётся через запятую;

<STRING> – текст, задаётся строкой от 1 до 255 символов;

all – удаляет все опции.

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IPV6-DHCP-SERVER

## Пример

```
esr(config-ipv6-dhcp-server)# option 43 ip-address fc00::/110
```

## show ipv6 dhcp binding

Данная команда позволяет посмотреть выданные DHCP-сервером IPv6-адреса.

### Синтаксис

```
show ipv6 dhcp binding [ <IPV6-ADDR> ]
```

### Параметры

<IPV6-ADDR> – IPv6-адрес, опциональный параметр, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF]. Если в команде задан параметр <IPV6-ADDR>, то будет отображена информация, связанная только с указанным адресом.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

## Пример

```
esr# show ipv6 dhcp binding
IPv6 address                               Lease expires at
-----
2001:db8:0:1::18                          2015-06-09 23:39:45
```

## show ipv6 dhcp server pool

Данная команда позволяет посмотреть настроенные пулы IPv6-адресов. При указании имени выводится информация только для заданного пула.

### Синтаксис

```
show ipv6 dhcp server pool [ <POOL_NAME> ]
```

### Параметры

<POOL\_NAME> – имя пула, опциональный параметр, задается строкой до 31 символа.

### Необходимый уровень привилегий

1

**Командный режим**

ROOT

**Пример**

```

esr# show ipv6 dhcp server pool lan_pool_ipv6
Name:                lan_pool_ipv6
Network:             2001:abcd:dcba::/120
Address-ranges:     2001:abcd:dcba::2-2001:abcd:dcba::10
Excluded-address-ranges: --
Addresses:          --
Dns-server:         --
Options:            --
Max lease time (d:h:m): 001:00:00
Default lease time (d:h:m): 000:12:00

```

**show ipv6 dhcp server vendor-specific**

Данная команда позволяет посмотреть настроенные DHCP опции 43 и 60.

**Синтаксис**

```
show ipv6 dhcp server vendor-specific
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример**

```

esr# show ipv6 dhcp server vendor-specific
Vendor ID      Vendor options
-----
ELTEX          0x0b0931302e312e39302e32

```

## 30 Настройка SLA

- Общие команды настройки SLA
  - ip sla
  - ip sla mode
  - ip sla responder cisco
  - ip sla responder cisco port
  - show ip sla configuration
  - show ip sla test configuration
  - show ip sla test history
  - show ip sla test statistics
  - show ip sla test status
- Настройка Eltex-SLA
  - authentication algorithm
  - authentication key-chain
  - authentication key-string
  - control-phase authentication algorithm
  - control-phase authentication key-id
  - control-phase authentication key-string
  - control-phase destination-port
  - control-phase retry
  - control-phase source-port
  - control-phase timeout
  - description
  - disallow-fragmentation
  - dscp
  - enable
  - frequency
  - history
  - ip sla logging
  - ip sla test
  - ip sla schedule
  - icmp-echo
  - icmp-jitter
  - ip sla key-chain
  - ip sla responder
  - ip sla responder eltex
  - ip sla responder eltex port
  - packet-size
  - thresholds delay
  - thresholds jitter
  - thresholds losses
  - timeout
  - timeout
  - ttl
  - udp-jitter
  - vrf

### Общие команды настройки SLA

#### ip sla

Данной командой активируется функционал IP-SLA (SLA-agent).

Использование отрицательной формы команды (no) отключает функционал IP-SLA.

**Синтаксис**

```
[no] ip sla
```

**Параметры**

Отсутствуют.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip sla
```

**ip sla mode**

Данной командой устанавливается режим работы SLA-agent. Команда доступна в случае, если на устройстве активирована действующая лицензия wiSLA.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
ip sla mode <MODE>
```

```
no ip sla mode
```

**Параметры**

<MODE> – режим работы SLA-agent. Принимает значения:

- eltex – работа SLA-agent в режиме разработанном компанией ELTEX;
- wisla-local – работа SLA-agent в режиме, разработанном компанией Wellink;
- wisla-remote – работа SLA-client в режиме, разработанном компанией Wellink.

**Значение по умолчанию**

eltex

**Необходимый уровень привилегий**

10

## Командный режим

CONFIG

### Пример

```
esr(config)# ip sla mode wisla
```

### **ip sla responder cisco**

Данной командой на интерфейсе активируется функционал SLA-responder для Cisco-SLA-agent.

Использование отрицательной формы команды (no) на интерфейсе деактивирует функционал SLA-responder для Cisco-SLA-agent.

### Синтаксис

```
[no] ip sla responder cisco
```

### Параметры

Отсутствуют

### Значение по умолчанию

Отключено

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP

CONFIG-L2TP

CONFIG-PPTP

CONFIG-VTI

CONFIG-IF-E1  
CONFIG-IF-MULTILINK  
CONFIG-LOOPBACK

### Пример

```
esr(config-if-gi)# ip sla responder cisco
```

### ip sla responder cisco port

Данной командой устанавливается UDP-порт, на котором будет идти прослушивание запросов от Cisco-SLA-agent.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ip sla responder cisco port <PORT>  
no ip sla responder cisco port
```

### Параметры

<PORT> – номер UDP-порта, задается в диапазоне [1..65535].

### Значение по умолчанию

Не ограничено

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-GRE  
CONFIG-IP4IP  
CONFIG-L2TP  
CONFIG-PPTP

CONFIG-VTI  
CONFIG-IF-E1  
CONFIG-IF-MULTILINK  
CONFIG-LOOPBACK

### Пример

```
esr(config-if-gi)# ip sla responder cisco port 7777
```

### show ip sla configuration

Данной командой выводится информация о текущих настройках SLA-agent.

### Синтаксис

```
show ip sla configuration
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
esr# show ip sla configuration
Mode:                               Eltex
State:                               Enabled
Logging:
  Delay:                             Enabled
  Error:                             Enabled
  Jitter:                            Enabled
  Losses:                            Enabled
  Status:                            Enabled
```

### show ip sla test configuration

Данной командой выводится информация о сконфигурированных SLA-тестах.

### Синтаксис

```
show ip sla test configuration [ <NUM> ] [ vrf <VRF> ]
```



## Параметры

<NUM> – номер SLA-теста, задается в диапазоне [1..10000];

<VRF> – имя VRF, задается строкой до 31 символа.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```

esr# show ip sla test configuration
Test      State      Source      Destination  Freq  Interval  Packets
PacketSize Timeout  Description
-----
1         Enabled    192.168.44.11  192.168.44.13  10    20        100
74                               DNS reachability
2         Enabled    192.168.44.11  192.168.44.1  10    20        100
74                               Channel to Gateway
3         Enabled    192.168.44.11  192.168.44.13  10    20        100
74                               Channel to DNS-server

```

```

esr# show ip sla test configuration 2

```

```

Test number:          2
Description:          Channel to Gateway
Test type:            udp-jitter
State:                Enabled
Authentication:      Disabled
Destination address:  192.168.44.13
Destination port:     1
Disallow fragmentation: Disabled
Frequency:            10
Interval:             20
Number of packets:    100
Packet size:          74
Source address:       192.168.44.1
Source interface:     --
Source port:          1
DSCP:                 0
Control phase destination port: 1800
Control phase source port: --
Timeout:              3000
Number of history records: 10

```

## show ip sla test history

Данной командой выводится информация об истории измерений SLA-теста.

**Синтаксис**

```
show ip sla test history [ <NUM> ] [ vrf <VRF> ] [ microseconds ] [ status ]
```

**Параметры**

<NUM> – номер SLA-теста, принимает значение [1..10000];

<VRF> – имя VRF, задается строкой до 31 символа;

microseconds – ключ для отображения истории измерений в микросекундах;

status – ключ для отображения истории статусов SLA-теста.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример**

```
esr# show ip sla test history 3
```

Probe	TP	LP	LPF	LPR	OWDFA	OWDRA	OWJFA	OWJRA	TWDA
TWJA	DP	OSF	OSR	Last Run					
1	100	0	0	0	0.17	0.17	0.01	0.01	0.35
0.03	0	0	0	2024-06-20 11:21:02					
2	100	0	0	0	0.17	0.17	0.01	0.01	0.35
0.03	0	0	0	2024-06-20 11:20:52					
3	100	0	0	0	0.17	0.17	0.01	0.01	0.35
0.03	0	0	0	2024-06-20 11:20:42					
4	100	0	0	0	0.17	0.17	0.01	0.01	0.34
0.03	0	0	0	2024-06-20 11:20:32					
5	100	0	0	0	0.17	0.17	0.01	0.01	0.35
0.03	0	0	0	2024-06-20 11:20:22					
6	100	0	0	0	0.18	0.18	0.01	0.01	0.35
0.03	0	0	0	2024-06-20 11:20:11					
7	100	0	0	0	0.17	0.17	0.01	0.01	0.34
0.03	0	0	0	2024-06-20 11:20:02					
8	100	0	0	0	0.17	0.17	0.01	0.01	0.34
0.03	0	0	0	2024-06-20 11:19:52					
9	100	0	0	0	0.17	0.17	0.01	0.01	0.34
0.03	0	0	0	2024-06-20 11:19:42					
10	100	0	0	0	0.17	0.17	0.01	0.01	0.34
0.03	0	0	0	2024-06-20 11:19:32					

**show ip sla test statistics**

Данной командой выводится информация об актуальных результатах измерений SLA-теста.

**Синтаксис**

```
show ip sla test statistics [ <NUM> ] [ vrf <VRF> ] [ microseconds ]
```

## Параметры

<NUM> – номер SLA-теста, принимает значение [1..10000];

<VRF> – имя VRF, задается строкой до 31 символа;

microseconds – ключ для отображения статистики в микросекундах.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```

esr# show ip sla test statistics
Test      TP      LP      LPF      LPR      OW DFA   OW DRA   OW JFA   OW JRA   TW DA
TWJA      DP      OSF      OSR
-----  -----  -----  -----  -----  -----  -----  -----  -----  -----
1         100     --      --      --      --      --      --      --      0.35
0.05     --      --      --
2         100     0       0       0       0.19    0.19    0.04    0.04    0.38
0.08     0       0       0
3         100     0       0       0       0.18    0.18    0.02    0.02    0.35
0.03     0       0       0

```

```

esr# show ip sla test statistics 2
Test number:                2
Description:                Channel to Gateway
Test status:                 Successful
Transmitted packets:        100
Lost packets:                0 (0.00%)
Lost packets in forward direction: 0 (0.00%)
Lost packets in reverse direction: 0 (0.00%)
One-way delay forward min/avg/max: 0.13/0.19/0.32 milliseconds
One-way delay reverse min/avg/max: 0.13/0.19/0.32 milliseconds
One-way jitter forward:      0.04 milliseconds
One-way jitter reverse:      0.04 milliseconds
Two-way delay min/avg/max:   0.27/0.38/0.64 milliseconds
Two-way jitter min/avg/max:  0.02/0.08/0.49 milliseconds
Duplicate packets:           0
Out of sequence packets in forward direction: 0
Out of sequence packets in reverse direction: 0
Number of successes:         10 (100.00%)
Number of failures:          0 (0.00%)

```

## show ip sla test status

Данной командой выводится сводная информация о статусах всех активных SLA-тестов.

## Синтаксис

```
show ip sla test status
```

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```

esr# show ip sla test status
Test      Type           Source           Destination      Status           Last Run
Description
-----
1         icmp-echo      192.168.44.11    192.168.44.13    Successful       2024-06-20 10:25:1
3         DNS reachability
2         udp-jitter     192.168.44.12    192.168.44.1     Successful       2024-06-20 10:25:1
4         Channel to Gateway
3         icmp-jitter    192.168.44.13    192.168.32.13    Successful       2024-06-20 10:25:1
1         Channel to DNS-server

```

## Настройка Eltex-SLA

### authentication algorithm

Данной командой указывается алгоритм для аутентификации входящих запросов от SLA-agent.

Использование отрицательной формы команды (no) удаляет алгоритм для аутентификации входящих запросов от SLA-agent.

### Синтаксис

```
authentication algorithm <ALGORITHM>
```

```
no authentication algorithm
```

### Параметры

<ALGORITHM> – алгоритм аутентификации: sha-256, hmac-sha-256.

### Значение по умолчанию

Не задан.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-SLA-RESPONDER

### Пример

```
esr(config-sla-responder)# authentication algorithm sha-256
```

### authentication key-chain

Данной командой указывается набор ключей для аутентификации входящих запросов от SLA-agent.

Использование отрицательной формы команды (no) удаляет набор ключей для аутентификации входящих запросов от SLA-agent.

### Синтаксис

```
authentication key-chain <KEYCHAIN>
```

```
no authentication key-chain
```

### Параметры

<KEYCHAIN> – идентификатор списка ключей, задается строкой до 16 символов.

### Значение по умолчанию

Не задан.

### Необходимый уровень привилегий

15

## Командный режим

CONFIG-SLA-RESPONDER

### Пример

```
esr(config-sla-responder)# authentication key-chain sla-chain
```

### authentication key-string

Данной командой указывается пароль для аутентификации входящих запросов от SLA-agent.

Использование отрицательной формы команды (no) удаляет пароль для аутентификации входящих запросов от SLA-agent.

### Синтаксис

```
authentication key-string ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
```

```
no authentication key-string
```

## Параметры

<CLEAR-TEXT> – пароль, задаётся строкой от 8 до 16 символов;

<ENCRYPTED-TEXT> – зашифрованный пароль размером от 8 байт до 16 байт (от 16 до 32 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-SLA-RESPONDER

## Пример

```
esr(config-sla-responder)# authentication key-string ascii-text aukey
```

## control-phase authentication algorithm

Данной командой задается алгоритм аутентификации для SLA-теста.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
control-phase authentication algorithm <ALGORITHM>
```

```
no control-phase authentication algorithm
```

## Параметры

<ALGORITHM> – алгоритм хеширования, принимает значения [sha-256, hmac-sha-256].

## Значение по умолчанию

Без шифрования

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-SLA-TEST

## Пример

```
esr(config-sla-test)# control-phase authentication algorithm hmac-sha-256
```

### control-phase authentication key-id

Данной командой выбирается номер ключа в key-chain, сконфигурированном командой ip sla key-chain (см. раздел [authentication key-chain](#)), который будет отправляться для аутентификации на SLA-responder.

Использование отрицательной формы команды (no) удаляет номер ключа.

#### Синтаксис

```
control-phase authentication key-id <ID>
```

```
no control-phase authentication key-id
```

#### Параметры

<ID> – идентификатор ключа, задается в диапазоне [1..255].

#### Необходимый уровень привилегий

15

#### Командный режим

CONFIG-SLA-TEST

#### Пример

```
esr(config-sla-test)# control-phase authentication key-id 2
```

### control-phase authentication key-string

Данной командой задается ключ аутентификации, который будет отправляться для аутентификации на SLA-responder.

Использование отрицательной формы команды (no) удаляет ключ аутентификации, который будет отправляться для аутентификации на SLA-responder.

#### Синтаксис

```
control-phase authentication key-string ascii-text { <CLEAR-TEXT> |encrypted <ENCRYPTED-TEXT>}
```

```
no control-phase authentication key-string
```

#### Параметры

<CLEAR-TEXT> – пароль, задаётся строкой от 8 до 16 символов;

<ENCRYPTED-TEXT> – зашифрованный пароль размером от 8 байт до 16 байт (от 16 до 32 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

#### Необходимый уровень привилегий

15

## Командный режим

CONFIG-SLA-TEST

### Пример

```
esr(config-sla-test)# control-phase authentication key-string ascii-text conphkey
```

### control-phase destination-port

Данной командой устанавливается UDP-порт, на который будут отправляться запросы контрольной фазы на Eltex-SLA-responder.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
control-phase destination-port <PORT>
```

```
no control-phase destination-port
```

### Параметры

<PORT> – номер UDP-порта, принимает значение [1..65535].

### Значение по умолчанию

1800

### Необходимый уровень привилегий

15

## Командный режим

CONFIG-SLA-TEST

### Пример

```
esr(config-sla-test)# control-phase destination-port 9999
```

### control-phase retry

Данной командой устанавливается периодичность попыток установления контрольной фазы SLA-теста.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
control-phase retry <TIME>
```

```
no control-phase retry
```



## Параметры

<TIME> – время ожидания перед повторной попыткой установления контрольной фазы SLA-теста, определяется в секундах [1..86400].

## Значение по умолчанию

1

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-SLA-TEST

## Пример

```
esr(config-sla-test)# control-phase retry 300
```

## control-phase source-port

Данной командой устанавливается UDP-порт, с которого будут отправляться запросы контрольной фазы на Eltex-SLA-responder.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
control-phase source-port <PORT>
```

```
no control-phase source-port
```

## Параметры

<PORT> – номер UDP-порта, принимает значение [1..65535].

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-SLA-TEST

## Пример

```
esr(config-sla-test)# control-phase source-port 3333
```

## control-phase timeout

Данной командой устанавливается время ожидания ответа от SLA-responder, по истечении которого контрольная фаза SLA-теста будет считаться неудачной.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
control-phase timeout <TIME>  
no control-phase timeout
```

### Параметры

<TIME> – время в миллисекундах, отведенное на прохождение контрольной фазы SLA-теста, принимает значение [1..4294967295].

### Значение по умолчанию

3000

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-SLA-TEST

### Пример

```
esr(config-sla-test)# control-phase timeout 5000
```

## description

Данной командой задается описание SLA-теста.

Использование отрицательной формы команды (no) удаляет описание SLA-теста.

### Синтаксис

```
description <DESCRIPTION>  
no description
```

### Параметры

<DESCRIPTION> – описание SLA-теста, задается строкой до 255 символов.

### Значение по умолчанию

Отсутствует

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-SLA-TEST

**Пример**

```
esr(config-sla-test)# description "Local gateway reachability"
```

**disallow-fragmentation**

Данная команда выставляет бит DF (don't fragment) в IP-заголовке равным единице для исходящих на SLA-responder пакетов.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
[no] disallow-fragmentation
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

DF бит не установлен ( равен 0)

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-SLA-TEST

**Пример**

```
esr(config-sla-test)# disallow-fragmentation
```

**dscp**

Данной командой задается значение DSCP для исходящих на SLA-responder пакетов.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
dscp <DSCP>
```

no dscp

### Параметры

<DSCP> – значение кода DSCP, принимает значение [0..63].

### Значение по умолчанию

0

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-SLA-TEST

### Пример

```
esr(config-sla-test)# dscp 58
```

### enable

Данной командой активируется SLA-тест.

Использование отрицательной формы команды (no) деактивирует SLA-тест.

### Синтаксис

[no] enable

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Выключен.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-SLA-TEST

### Пример

```
esr(config-sla-test)# enable
```

**frequency**

Данной командой устанавливается период между перезапусками SLA-теста.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
frequency <TIME>
```

```
no frequency
```

**Параметры**

<TIME> – время в секундах, принимает значение [1..604800].

**Значение по умолчанию**

10

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-SLA-TEST

**Пример**

```
esr(config-sla-test)# frequency 3600
```

**history**

Данной командой задается количество сохраняемых результатов SLA-теста.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
history <SIZE>
```

```
no history
```

**Параметры**

<SIZE> – число сохраняемых результатов, принимает значение [1..1000].

**Значение по умолчанию**

10

**Необходимый уровень привилегий**

10

## Командный режим

CONFIG-SLA-TEST

### Пример

```
esr(config-sla-test)# history 100
```

### ip sla logging

Данной командой активируется логирование групп событий SLA.

Использование отрицательной формы команды (no) отключает логирование групп событий SLA.

### Синтаксис

```
[no] ip sla logging <TYPE>
```

### Параметры

<TYPE> – название группы информационных сообщений, может принимать значения:

- error – отображение сообщений об ошибках в работе SLA-тестов, причинах их провала, а также ошибок в работе SLA-responder (если таковой сконфигурирован в системе);
- delay – отображение сообщений о превышении/нормализации значений, установленных в thresholds delay;
- jitter – отображение сообщений о превышении/нормализации значений, установленных в thresholds jitter;
- losses – отображение сообщений о превышении/нормализации значений, установленных в thresholds losses;
- status – отображение сообщений о смене статуса SLA-теста.

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG

### Пример

```
esr(config)# ip sla logging status  
esr(config)# ip sla logging error
```

### ip sla test

Данная команда используется для создания SLA-теста (SLA-agent) и перехода в режим его конфигурирования.

Использование отрицательной формы команды (no) удаляет указанный SLA-тест.

### Синтаксис

```
[no] ip sla test <NUM>
```

### Параметры

<NUM> – номер SLA-теста, принимает значение [1..10000].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# ip sla test 10
```

### ip sla schedule

Данной командой устанавливается планировщик расписания работы SLA-тестов. Если SLA-тест создан и активирован, для него необходимо задать расписание.

Использование отрицательной формы команды (no) удаляет расписание работы SLA-тестов.

### Синтаксис

```
ip sla schedule { <TEST-NUMBER> | all } [ life { <LIFE-TIME> | forever } ] [ start-time { <TIME> <MONTH> <DAY> | now } ]
```

```
[no] ip sla schedule { <TEST-NUMBER> | all }
```

### Параметры

<TEST-NUMBER> – номер SLA-теста, принимает значение [1..10000]. При использовании ключа "all" вместо номера, расписание устанавливается для всех созданных SLA-тестов;

<LIFE-TIME> – время жизни теста, принимает значение [1..2147483647] секунд;

forever – время жизни теста не ограничено;

<TIME> – время начала теста, задаётся в виде HH:MM:SS, где:

- HH – часы, принимает значение [0..23];
- MM – минуты, принимает значение [0..59];
- SS – секунды, принимает значение [0..59].

<MONTH> – месяц начала теста, принимает значения [ January / February / March / April / May / June / July / August / September / October / November / December ];

<DAY> – день месяца начала теста, принимает значения [1..31];

now – начать тест немедленно.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# ip sla schedule 1 life forever start-time now
```

## icmp-echo

Данной командой задаётся icmp-режим тестирования канала связи и параметры тестирования.

Использование отрицательной формы команды (no) удаляет конфигурацию режима и параметров тестирования.

## Синтаксис

```
icmp-echo <DST-ADDRESS> { source-ip { <SRC-ADDRESS> | object-group
<NETWORK_OBJ_GROUP_NAME> } | source-interface { <IF> | <TUN> } } [ interval <INTERVAL> ]
[ num-packets <NUM-PACKETS> ]
```

```
no icmp-echo
```

## Параметры

<DST-ADDRESS> – IPv4-адрес SLA-responder. IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<SRC-ADDRESS> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<NETWORK\_OBJ\_GROUP\_NAME> – список адресов, которые будут использоваться в качестве source address;

<IF> – тип и идентификатор интерфейса/туннеля, IP-адрес которого будет использоваться в качестве адреса источника пакетов, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#);

<INTERVAL> – интервал отправки тестовых пакетов, принимает значение [1..6000] миллисекунд;

<NUM-PACKETS> – количество тестовых пакетов, отправляемых в рамках теста, принимает значение [1..100000].

## Значение по умолчанию

<INTERVAL> – 20;

<NUM-PACKETS> – 100.



**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-SLA-TEST

**Пример**

```
esr(config-sla-test)# icmp-echo 10.10.1.1 source-ip 192.168.54.22 num-packets 50 interval 15
```

**icmp-jitter**

Данной командой задаётся icmp-режим тестирования качества канала связи и параметры тестирования.

Использование отрицательной формы команды (no) удаляет конфигурацию режима и параметров тестирования.

**Синтаксис**

```
icmp-jitter <DST-ADDRESS> { source-ip <SRC-ADDRESS> | source-interface { <IF> | <TUN> } }
[ interval <INTERVAL> ] [ num-packets <NUM-PACKETS> ]
```

```
no icmp-echo
```

**Параметры**

<DST-ADDRESS> – IPv4-адрес SLA-responder. IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<SRC-ADDRESS> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IF> – тип и идентификатор интерфейса/туннеля, IP-адрес которого будет использоваться в качестве адреса источника пакетов, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#);

<INTERVAL> – интервал отправки тестовых пакетов, принимает значение [1..6000] миллисекунд;

<NUM-PACKETS> – количество тестовых пакетов, отправляемых в рамках теста, принимает значение [1..100000].

**Значение по умолчанию**

<INTERVAL> – 20;

<NUM-PACKETS> – 100.

**Необходимый уровень привилегий**

10

## Командный режим

CONFIG-SLA-TEST

### Пример

```
esr(config-sla-test)# icmp-jitter 10.10.1.1 source-ip 192.168.54.22 num-packets 300 interval 100
```

### ip sla key-chain

Данная команда определяет набор паролей для аутентификации через алгоритм хеширования md5 с SLA-responder/agent.

Использование отрицательной формы команды (no) удаляет привязку к набору паролей.

### Синтаксис

```
ip sla key-chain <NAME>
```

```
no ip sla key-chain
```

### Параметры

<NAME> – имя, задаётся строкой от 1 до 16 символов.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG

### Пример

```
esr(config)# ip sla key-chain KEY10
```

### ip sla responder

Данной командой выполняется переход в режим конфигурирования параметров SLA-responder.

Использование отрицательной формы команды (no) удаляет примененные ранее параметры SLA-responder.

### Синтаксис

```
[no] ip sla responder [ vrf <VRF> ]
```

### Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра SLA-responder включается в указанном VRF.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# ip sla responder
```

**ip sla responder eltex**

Данной командой на интерфейсе активируется функционал SLA-responder для Eltex-SLA-agent.

Использование отрицательной формы команды (no) на интерфейсе деактивируется функционал SLA-responder для Eltex-SLA-agent.

**Синтаксис**

```
[no] ip sla responder eltex
```

**Параметры**

Отсутствуют.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP

CONFIG-L2TP

CONFIG-PPTP  
CONFIG-VTI  
CONFIG-IF-E1  
CONFIG-IF-MULTILINK  
CONFIG-LOOPBACK

### Пример

```
esr(config)# ip sla responder eltex
```

### ip sla responder eltex port

Данной командой устанавливается UDP-порт, на котором будет идти прослушивание запросов от Eltex-SLA-agent.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

### Синтаксис

```
ip sla responder eltex port <PORT>  
no ip sla responder eltex port
```

### Параметры

<PORT> – номер UDP-порта, задается в диапазоне [1..65535].

### Значение по умолчанию

1800

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IF-GI  
CONFIG-IF-TE  
CONFIG-IF-TWE  
CONFIG-IF-FO  
CONFIG-IF-HU  
CONFIG-IF-SUB  
CONFIG-IF-QINQ  
CONFIG-IF-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-GRE  
CONFIG-IP4IP

CONFIG-L2TP  
CONFIG-PPTP  
CONFIG-VTI  
CONFIG-IF-E1  
CONFIG-IF-MULTILINK  
CONFIG-LOOPBACK

### Пример

```
esr(config)# ip sla responder eltex port 5555
```

### packet-size

Данной командой задается размер исходящих на SLA-responder пакетов.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
packet-size <SIZE>  
no packet-size
```

### Параметры

<SIZE> – размер тестовых пакетов SLA-теста, принимает значение [70..18000] байт.

### Значение по умолчанию

74

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-SLA-TEST

### Пример

```
esr(config-sla-test)# packet-size 256
```

### thresholds delay

Данной командой устанавливаются допустимые значения задержек, при пересечении которых тест считается проваленным.

Использование отрицательной формы команды (no) удаляет пороги и отключает отслеживание параметров задержки.

**Синтаксис**

```
thresholds delay { high <DELAY> | low <DELAY> | forward { high <DELAY> | low <DELAY> } |
reverse {high <DELAY> | low <DELAY> } }
no thresholds delay { high | low | forward { high | low } | reverse { high | low } }
```

**Параметры**

high – максимально допустимое значение параметра;

low – минимально допустимое значение параметра;

forward – измерение параметра в направлении от SLA-agent до SLA-responder;

reverse – измерение параметра в направлении от SLA-responder до SLA-agent;

Без указания forward/reverse – задаются суммарные параметры задержки;

<DELAY> – значение задержки в миллисекундах, принимает значение в диапазоне [1..60000].

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-SLA-TEST

**Пример**

```
esr(config-sla-test)# thresholds delay low 20
```

**thresholds jitter**

Данной командой устанавливаются допустимые значения джиттера, при пересечении которых тест считается проваленным.

Использование отрицательной формы команды (no) удаляет пороги и отключает отслеживание параметров задержки.

**Синтаксис**

```
thresholds jitter { high <JITTER> | low <JITTER> | forward { high <JITTER> | low
<JITTER> } | reverse { high <JITTER> | low <JITTER> } }
no thresholds jitter { high | low | forward { high | low } | reverse { high | low } }
```

**Параметры**

high – максимально допустимое значение параметра;

low – минимально допустимое значение параметра;

forward – измерение параметра в направлении от SLA-agent до SLA-responder;

reverse – измерение параметра в направлении от SLA-responder до SLA-agent;

Без указания forward/reverse – задаются суммарные параметры джиттера;

<JITTER> – значение джиттера в миллисекундах, принимает значение в диапазоне [1..60000].

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-SLA-TEST

### Пример

```
esr(config-sla-test)# thresholds jitter forward high 10
```

### thresholds losses

Данной командой устанавливаются допустимые значения потерь, при пересечении которых тест считается проваленным.

Использование отрицательной формы команды (no) удаляет пороги и отключает отслеживание параметров потерь.

### Синтаксис

```
thresholds losses { high <NUM-PACKETS> | low <NUM-PACKETS> | forward { high <NUM-PACKETS> | low <NUM-PACKETS> } | reverse { high <NUM-PACKETS> | low <NUM-PACKETS> } }
no thresholds losses { high | low | forward | low } | reverse { high | low } }
```

### Параметры

high – максимально допустимое значение параметра;

low – минимально допустимое значение параметра;

forward – измерение параметра в направлении от SLA-agent до SLA-responder;

reverse – измерение параметра в направлении от SLA-responder до SLA-agent;

Без указания forward/reverse – задаются суммарные параметры потерь;

<NUM-PACKETS> – количество пакетов, принимает значение в диапазоне [1..60000].

### Значение по умолчанию

Отключено.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-SLA-TEST

### Пример

```
esr(config-sla-test)# thresholds losses reverse high 10
```

### timeout

Данной командой устанавливается время ожидания очередного пакета от SLA-responder, по истечении которого SLA-agent закрывает SLA-сессию.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

### Синтаксис

```
timeout <TIME>
```

```
no timeout
```

### Параметры

<TIME> – время ожидания, определяется в миллисекундах [1..4294967295].

### Значение по умолчанию

3000

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-SLA-TEST

### Пример

```
esr(config-sla-test)# timeout 1000
```

### timeout

Данной командой устанавливается время ожидания очередного пакета от SLA-agent, по истечении которого SLA-responder закрывает SLA-сессию.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

### Синтаксис

```
timeout <TIME>
```

```
no timeout
```



**Параметры**

<TIME> – время, отведенное на прохождение SLA-теста, определяется в миллисекундах [1..4294967295].

**Значение по умолчанию**

3000

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-SLA-RESPONDER

**Пример**

```
esr(config-sla-responder)# timeout
```

**ttl**

Данной командой задается значение TTL для исходящих на SLA-responder пакетов.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

**Синтаксис**

```
ttl <TTL>
```

```
no ttl
```

**Параметры**

<TTL> – значение TTL, принимает значения в диапазоне [1..255].

**Значение по умолчанию**

64

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-SLA-TEST

**Пример**

```
esr(config-sla-test)# ttl 10
```

## udp-jitter

Данной командой задаётся UDP-режим тестирования канала связи и параметры тестирования.

Использование отрицательной формы команды (no) удаляет конфигурацию режима и параметров тестирования.

### Синтаксис

```
udp-jitter <DST-ADDRESS> <DST-PORT> { source-ip { <SRC-ADDRESS> | object-group
<NETWORK_OBJ_GROUP_NAME> } | source-interface { <IF> | <TUN> } } [ source-port <SRC-
PORT> ] [ interval <INTERVAL> ] [ num-packets <NUM-PACKETS> ]
```

```
no udp-jitter
```

### Параметры

<DST-ADDRESS> – IPv4-адрес SLA-responder. IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<DST-PORT> – номер UDP-порта назначения тестовых пакетов, принимает значения [1..65535];

<SRC-ADDRESS> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<NETWORK\_OBJ\_GROUP\_NAME> – список адресов, которые будут использоваться в качестве source address;

<IF> – тип и идентификатор интерфейса/туннеля, IP-адрес которого будет использоваться в качестве адреса источника пакетов, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#);

<SRC-PORT> – номер UDP-порта источника тестовых пакетов, принимает значения [1..65535];

<INTERVAL> – интервал отправки тестовых пакетов, принимает значение [1..255] миллисекунд;

<NUM-PACKETS> – количество тестовых пакетов, отправляемых в рамках теста, принимает значение [1..6000];

<SRC-ADDRESS> – source-адрес тестовых пакетов SLA-теста, задается IPv4-адресом. Может принимать значение IPv4-адреса.

### Значение по умолчанию

control enable

<INTERVAL> – 20

<NUM-PACKETS> – 100

<SRC-PORT> – любой свободный

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-SLA-TEST

## Пример

```
esr(config-sla-test)# udp-jitter 10.10.1.1 50000 source-ip 192.168.54.22 num-packets 50  
interval 15
```

## vrf

Данной командой задается экземпляр VRF, в адресном пространстве которого должен работать SLA-тест.

Использование отрицательной формы команды (no) возвращает SLA-тест в адресное пространство глобального маршрутизатора.

## Синтаксис

```
vrf <VRF>
```

```
no vrf
```

## Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

## Необходимый уровень привилегий

10

## Командный режим


CONFIG-SLA-TEST

## Пример

```
esr(config-sla-test)# vrf subrouter
```

## 31 Настройка контроля абонентов (BRAS)

- `aaa das-profile`
- `aaa services-radius-profile`
- `aaa sessions-radius-profile`
- `backup traffic-processing transparent`
- `bypass-traffic-acl`
- `class-map`
- `clear subscriber-control sessions`
- `default-action`
- `default-service`
- `description`
- `dhcp-option-82-include accept-time`
- `dhcp-option-82-include enable`
- `dhcp-option-82-include lease-time`
- `dhcp-option-82-include size`
- `enable`
- `failover`
- `filter-action`
- `filter-name`
- `ip proxy http listen-ports`
- `ip proxy https listen-ports`
- `ip proxy source-address`
- `location`
- `nas-interface`
- `nas-ip-address`
- `quota-expired-reauth`
- `service-subscriber-control`
- `session accounting`
- `session ip-authentication`
- `session ip-authentication password`
- `session l2-roaming disable`
- `session l2-roaming realtime-accounting`
- `session mac-authentication`
- `session mac-authentication password`
- `session unknown-mac-address`
- `session-timeout`
- `show subscriber-control configuration`
- `show subscriber-control radius-servers`
- `show subscriber-control services count`
- `show subscriber-control services counters`
- `show subscriber-control services status`
- `show subscriber-control sessions count`
- `show subscriber-control sessions counters`
- `show subscriber-control sessions status`
- `subscriber-control`
- `subscriber-control application-filter`
- `subscriber-control apps-server-url`
- `subscriber-control filters-server-url`
- `subscriber-control peer-address`
- `subscriber-control thresholds sessions-number`
- `subscriber-control unused-filters-remove-delay`
- `thresholds sessions-number`
- `vrrp-group`

 Данный функционал поддерживается при наличии лицензии.

## aaa das-profile

Данная команда используется для выбора профиля серверов динамической авторизации (DAS), на которые будут приходить CoA-запросы от PCRF об изменении политики обслуживания, а также запросы оперативной информации от CaptivePortal.

Использование отрицательной формы команды (no) удаляет заданный профиль серверов динамической авторизации (DAS).

### Синтаксис

```
[no] aaa das-profile <NAME>
```

### Параметры

<NAME> – имя профиля серверов динамической авторизации (DAS), задается строкой до 31 символа.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-SUBSCRIBER-CONTROL

### Пример

```
esr(config-subscriber-control)# aaa das-profile profile1
```

## aaa services-radius-profile

Данная команда используется для выбора профиля RADIUS-серверов, на которые будут отправляться запросы для получения параметров сервиса пользователя. Если профиль не задан, то будет использоваться профиль «aaa sessions-radius-profile».

Использование отрицательной формы команды (no) удаляет заданный профиль RADIUS-серверов.

### Синтаксис

```
[no] aaa services-radius-profile <NAME>
```

### Параметры

<NAME> – имя профиля RADIUS-серверов, задается строкой до 31 символа.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-SUBSCRIBER-CONTROL

## Пример

```
esr(config-subscriber-control)# aaa services-radius-profile profile1
```

## aaa sessions-radius-profile

Данная команда используется для выбора профиля RADIUS-серверов, на которые будут отправляться запросы для получения параметров сессии пользователя.

Использование отрицательной формы команды (no) удаляет заданный профиль RADIUS-серверов.

## Синтаксис

```
[no] aaa sessions-radius-profile <NAME>
```

## Параметры

<NAME> – имя профиля RADIUS-серверов, задается строкой до 31 символа.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-SUBSCRIBER-CONTROL

## Пример

```
esr(config-subscriber-control)# aaa sessions-radius-profile profile1
```

## backup traffic-processing transparent

Данная команда используется для включения прозрачного пропускания трафика в состоянии backup для BRAS.

Использование отрицательной формы команды (no) отключает прозрачное пропускание трафика в состоянии backup для BRAS.

## Синтаксис

```
[no] backup traffic-processing transparent
```

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-SUBSCRIBER-CONTROL

**Пример**

```
esr(config-subscriber-control)# backup traffic-processing transparent
```

**bypass-traffic-acl**

Данная команда используется для организации прозрачного пропускания служебного трафика (DHCP, DNS и т. д.) на основе фильтров.

Использование отрицательной формы команды (no) отключает прозрачное пропускание трафика.

**Синтаксис**

```
bypass-traffic-acl <NAME>
```

```
no bypass-traffic-acl
```

**Параметры**

<NAME> – имя привязываемого ACL, задается строкой до 31 символа.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-SUBSCRIBER-CONTROL

**Пример**

```
esr(config-subscriber-control)# bypass-traffic-acl LANs
```

**class-map**

Данная команда используется для привязки указанного QoS-класса к сервису по умолчанию. Прохождение трафика, не входящего в QoS-класс, запрещено.

Использование отрицательной формы команды (no) удаляет привязку класса к сервису по умолчанию.

**Синтаксис**

```
[no] class-map <NAME>
```

## Параметры

<NAME> – имя привязываемого класса, задается строкой до 31 символа.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-SUBSCRIBER-DEFAULT-SERVICE

## Пример

```
esr(config-subscriber-default-service)# class-map LAN
```

## clear subscriber-control sessions

Данной командой осуществляется удаление активных сессий контроля пользователей.

## Синтаксис

```
clear subscriber-control sessions [ vrf <VRF> ] [ username <USER-NAME> ] [ session-id <SESSION-ID> ]
```

## Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа. При указании данного параметра будут удалены активные сессии в указанном VRF;

<USER-NAME> – имя пользователя, задается строкой до 230 символов;

<SESSION-ID> – идентификатор сессии, принимает значения [1..18446744073709551615].

## Необходимый уровень привилегий

10

## Командный режим

ROOT

## Пример

```
esr# clear subscriber-control sessions
```

## default-action

Данная команда используется для указания действия, которое должно быть применено для HTTP/HTTPS-пакетов, URL (HTTP Host для HTTPS-пакетов), в которых не входит в список URL, назначенный командой «filter-name» (см. раздел [filter-name](#)).

Использование отрицательной формы команды (no) удаляет назначенное действие.



**Синтаксис**

```
default-action <АСТ>
no default-action
```

**Параметры**

<АСТ> – назначаемое действие:

- permit – прохождение трафика разрешается;
- deny – прохождение трафика запрещается;
- redirect <URL> – будет выполнен редирект на указанный URL, задается строкой до 255 символов.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-SUBSCRIBER-DEFAULT-SERVICE

**Пример**

```
esr(config-subscriber-default-service)# default-action redirect http://192.162.1.2/cp
```

**default-service**

Переход в режим конфигурирования сервиса по умолчанию. Сервис по умолчанию применяется для всех новых пользовательских сессий. После прохождения аутентификации пользователю назначаются персональные сервисы.

**Синтаксис**

```
default-service
```

**Параметры**

Команда не содержит параметров.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-SUBSCRIBER-CONTROL

**Пример**

```
esr(config-subscriber-control)# default-service
```

## description

Данной командой определяется описание профиля контроля пользователей.  
Использование отрицательной формы команды (no) удаляет описание.

### Синтаксис

```
description <DESCRIPTION>  
no description
```

### Параметры

<DESCRIPTION> – описание правила wan, задаётся строкой до 255 символов.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-SUBSCRIBER-CONTROL

### Пример

```
esr(config-subscriber-control)# description "Wi-Fi BRAS"
```

## dhcp-option-82-include accept-time

Данная команда устанавливает время, через которое неподтвержденная опция будет удалена.  
Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
[no] dhcp-option-82-include accept-time <SEC>
```

### Параметры

<SEC> – период времени в секундах, принимает значения [10..3600].

### Значение по умолчанию

10 секунд.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-SUBSCRIBER-CONTROL

## Пример

```
esr(config-subscriber-control)# dhcp-option-82-include accept-time 30
```

### **dhcp-option-82-include enable**

Данная команда активирует поиск option 82 в пакетах DHCP.

Использование отрицательной формы (no) деактивирует поиск option 82 в пакетах DHCP.

### **Синтаксис**

```
[no] enable
```

### **Параметры**

Команда не содержит параметров.

### **Значение по умолчанию**

Процесс выключен.

### **Необходимый уровень привилегий**

15

### **Командный режим**

CONFIG-SUBSCRIBER-CONTROL

## Пример

```
esr(config-subscriber-control)# enable
```

### **dhcp-option-82-include lease-time**

Данная команда устанавливает время, через которое запись об подтвержденной option 82 будет считаться устаревшей.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### **Синтаксис**

```
[no] dhcp-option-82-include lease-time <SEC>
```

### **Параметры**

<SEC> – период времени в секундах, принимает значения [60..86400].

### **Значение по умолчанию**

3600 секунд.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-SUBSCRIBER-CONTROL

**Пример**

```
esr(config-subscriber-control)# dhcp-option-82-include lease-time 7200
```

**dhcp-option-82-include size**

Данная команда устанавливает максимальный размер таблицы для хранения option 82.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
[no] dhcp-option-82-include size <SIZE>
```

**Параметры**

<SIZE> – размер таблицы option 82 [100..100000].

**Значение по умолчанию**

100 – для ESR-1x;

1000 – для ESR-20/30/100/200;

10000 – для ESR-1000/1200/1510/1511;

50000 – для ESR-1700/3100/3200.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-SUBSCRIBER-CONTROL

**Пример**

```
esr(config-subscriber-control)# dhcp-option-82-include size 25000
```

**enable**

Данной командой активируется профиль контроля пользователей.

Использование отрицательной формы команды (no) деактивирует профиль контроля пользователей.

## Синтаксис

[no] enable

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Процесс выключен.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-SUBSCRIBER-CONTROL

## Пример

```
esr(config-subscriber-control)# enable
```

## failover

Данная команда активирует работу резервирования BRAS. Указание адресации и группы описано в разделе [«Резервирование»](#).

Использование отрицательной формы команды (no) выключает работу резервирования BRAS.

Синтаксис

[no] failover

## Параметры

Отключено.

Необходимый уровень привилегий

10

## Командный режим

CONFIG-SUBSCRIBER-CONTROL

## Пример

```
wlc(config-subscriber-control)# failover
```

## filter-action

Данная команда используется для указания действия, которое должно быть применено для HTTP/HTTPS-пакетов, URL (HTTP Host для HTTPS-пакетов), в которых входит в список URL, назначенный командой «filter-name» (см. раздел [filter-name](#)).

Использование отрицательной формы команды (no) удаляет назначенное действие.

### Синтаксис

```
filter-action <ACT>
```

```
no filter-action
```

### Параметры

<ACT> – назначаемое действие:

- permit – прохождение трафика разрешается;
- deny – прохождение трафика запрещается;
- redirect <URL> – будет выполнен редирект на указанный URL, задается строкой до 255 символов.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-SUBSCRIBER-DEFAULT-SERVICE

### Пример

```
esr(config-subscriber-default-service)# filter-action redirect http://192.162.1.2/forbidden
```

## filter-name

Данная команда используется для указания имени списка URL, который будет использоваться для фильтрации HTTP/HTTPS-трафика не аутентифицированных пользователей. Список может быть настроен локально с помощью профиля URL либо получен с удаленного сервера (см. раздел [subscriber-control application-filter](#)).

Использование отрицательной формы команды (no) удаляет имя списка.

### Синтаксис

```
filter-name { local <LOCAL-NAME> | remote <REMOTE-NAME> }
```

```
no filter-name
```

### Параметры

<LOCAL-NAME> – имя профиля URL, задается строкой до 31 символа;

<REMOTE-NAME> – имя списка URL на удаленном сервере, задается строкой до 31 символа.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-SUBSCRIBER-DEFAULT-SERVICE

**Пример**

```
esr(config-subscriber-default-service)# filter-name local BLACK_LIST
```

**ip proxy http listen-ports**

Данной командой определяется с каких TCP-портов назначения трафик будет перенаправлен на HTTP Проxy-сервер маршрутизатора.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
ip proxy http listen-ports <NAME>
```

```
no ip proxy http listen-ports
```

**Параметры**

<NAME> – имя профиль TCP/UDP-портов, задаётся строкой до 31 символа.

**Значение по умолчанию**

80, 8080

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-SUBSCRIBER-CONTROL

**Пример**

```
esr(config-subscriber-control)# ip proxy http listen-ports HTTP_PORTS
```

**ip proxy https listen-ports**

Данной командой определяется с каких TCP-портов назначения трафик будет перенаправлен на HTTPS Проxy-сервер маршрутизатора.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
ip proxy https listen-ports <NAME>  
no ip proxy https listen-ports
```

**Параметры**

<NAME> – имя профиль TCP/UDP-портов, задаётся строкой до 31 символа.

**Значение по умолчанию**

443, 8443

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-SUBSCRIBER-CONTROL

**Пример**

```
esr(config-subscriber-control)# ip proxy https listen-ports HTTPS_PORTS
```

**ip proxy source-address**

Данной командой определяется IP-адрес маршрутизатора, который будет использоваться в качестве IP-адреса источника в отправляемых Proxy-сервером HTTP/HTTPS-пакетах.

Использование отрицательной формы команды (no) удаляет указанный IP-адрес источника.

**Синтаксис**

```
ip proxy source-address <ADDR>  
no ip proxy source-address
```

**Параметры**

<ADDR> – IP-адрес источника, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

**Значение по умолчанию**

По умолчанию используется IP-адрес интерфейса, с которого будет отправляться пакет.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-SUBSCRIBER-CONTROL



## Пример

```
esr(config-subscriber-control)# ip proxy source-address 10.100.100.2
```

## location

Данная команда используется для изменения идентификатора сетевого интерфейса. Данный идентификатор используется при HTTP-редиректах на CaptivePortal, а также передается в учетной информации RADIUS и при экспорте информации через Netflow-протокол.

Использование отрицательной формы команды (no) удаляет идентификатор.

## Синтаксис

```
location <ID>
```

```
no location
```

## Параметры

<ID> – идентификатор сетевого интерфейса, задаётся строкой до 220 символов.

## Значение по умолчанию

Отключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-L2TPV3

## Пример

```
esr(config-if-gi)# location "Guest SSID"
```

## nas-interface

Данной командой определяется интерфейс маршрутизатора, IP-адрес которого будет использоваться в качестве IP-адреса источника в отправляемых RADIUS-пакетах.

Использование отрицательной формы команды (no) удаляет указанный IP-адрес источника.

### Синтаксис

```
nas-interface {<IF> | <TUN>}  
no nas-interface
```

### Параметры

<IF> – интерфейс, задаётся в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#);

<TUN> – имя туннеля, задаётся в виде, описанном в разделе [Типы и порядок именования туннелей маршрутизатора](#).

### Значение по умолчанию

Не задано.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-SUBSCRIBER-CONTROL

### Пример

```
esr(config-subscriber-control)# nas-interface gi 1/0/1
```

## nas-ip-address

Данной командой определяется IP-адрес маршрутизатора, который будет использоваться в качестве IP-адреса источника в отправляемых RADIUS-пакетах.

Использование отрицательной формы команды (no) удаляет указанный IP-адрес источника.

### Синтаксис

```
nas-ip-address <ADDR>  
no nas-ip-address
```

### Параметры

<ADDR> – IP-адрес источника, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-SUBSCRIBER-CONTROL

## Пример

```
esr(config-subscriber-control)# nas-ip-address 10.100.100.2
```

## quota-expired-reauth

Данной командой можно включить перезапрос значения квоты при ее истечении для сервисов пользователя с настроенным ограничением по объему трафика или времени. В ином случае после истечения квоты сервис будет деактивирован, и пользователю будет назначен сервис по умолчанию. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

[no] quota-expired-reauth

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

При истечении квоты пользователю будет назначен сервис по умолчанию.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-SUBSCRIBER-CONTROL

## Пример

```
esr(config-subscriber-control)# quota-expired-reauth
```

## service-subscriber-control

Данная команда используется для включения контроля пользователей на интерфейсе. При выполнении со значением параметра «any» контроль пользователей будет работать для пакетов из любой подсети, в ином случае только для пакетов из подсетей, указанных в профиле IP-адресов.

Использование отрицательной формы команды (no) выключает контроль пользователей на интерфейсе.

**Синтаксис**

```
service-subscriber-control { any | object-group <NAME> }
no service-subscriber-control
```

**Параметры**

<NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа.

**Необходимый уровень привилегий**

15

**Командный режим**

```
CONFIG-IF-GI
CONFIG-IF-TE
CONFIG-IF-TWE
CONFIG-IF-FO
CONFIG-IF-HU
CONFIG-IF-SUB
CONFIG-IF-QINQ
CONFIG-IF-PORT-CHANNEL
CONFIG-BRIDGE
CONFIG-IP4IP4
CONFIG-GRE
```

**Пример**

```
esr(config-if-gi)# service-subscriber-control object-group LAN
```

**session accounting**

Данной командой устанавливается режим отправки сообщений RADIUS-accounting.

При использовании отрицательной формы команды (no) устанавливается значение по умолчанию.

**Синтаксис**

```
session accounting { all | auth-only }
```

**Параметры**

all – отправка для всех сессий;

auth-only – отправка для авторизированных сессий.

**Значение по умолчанию**

all

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-SUBSCRIBER-CONTROL

**Пример**

```
esr(config-subscriber-control)# session accounting all
```

**session ip-authentication**

Данной командой можно включить аутентификацию сессий по IP-адресу.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
[no] session ip-authentication
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-SUBSCRIBER-CONTROL

**Пример**

```
esr(config-subscriber-control)# session ip-authentication
```

**session ip-authentication password**

Данной командой возможно задать аутентификацию сессий по паролю.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
session ip-authentication password { <PASSWORD> | encrypted <ENCRYPTED-TEXT> }
no session ip-authentication
```

**Параметры**

<PASSWORD> – строка [1..64] ASCII-символов;

<ENCRYPTED-TEXT> – зашифрованный пароль размером [2..64] байт, задается строкой [2..128] символов.

**Значение по умолчанию**

По умолчанию пароль равен логину.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-SUBSCRIBER-CONTROL

**Пример**

```
esr(config-subscriber-control)# session ip-authentication password userpassword
```

**session l2-roaming disable**

Данной командой можно выключить режим прозрачного роуминга абонентов между L2-интерфейсами BRAS. В случае изменения L2-интерфейса абоненту необходимо будет заново пройти аутентификацию.

Использование отрицательной формы команды (no) включает режим прозрачного роуминга абонентов между L2-интерфейсами BRAS.

**Синтаксис**

```
[no] session l2-roaming disable
```

**Параметры**

Команда не содержит параметров.

**Значение по умолчанию**

Включен режим прозрачного роуминга абонентов между L2-интерфейсами BRAS.

**Необходимый уровень привилегий**

15

## Командный режим

CONFIG-SUBSCRIBER-CONTROL

### Пример

```
esr(config-subscriber-control)# session l2-roaming disable
```

### session l2-roaming realtime-accounting

Данной командой можно включить режим отправки RADIUS-аккаунтинга в реальном времени при изменении L2-интерфейса BRAS, по которому приходит работа с абонентом.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
[no] session l2-roaming realtime-accounting
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

RADIUS-аккаунтинг с измененным L2-интерфейсом отправляется по истечению Interim-Update.

### Необходимый уровень привилегий

15

## Командный режим

CONFIG-SUBSCRIBER-CONTROL

### Пример

```
esr(config-subscriber-control)# session l2-roaming realtime-accounting
```

### session mac-authentication

Данной командой можно включить аутентификацию сессий по MAC-адресу.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
[no] session mac-authentication
```

### Параметры

Команда не содержит параметров.

**Значение по умолчанию**

Отключено.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-SUBSCRIBER-CONTROL

**Пример**

```
esr(config-subscriber-control)# session mac-authentication
```

**session mac-authentication password**

Данной командой возможно задать аутентификацию сессий по паролю.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
session mac-authentication password { <PASSWORD> | encrypted <ENCRYPTED-TEXT> }  
[no] session mac-authentication
```

**Параметры**

<PASSWORD> – строка [1..64] ASCII-символов;

<ENCRYPTED-TEXT> – зашифрованный пароль, размером [2..64] байт, задаётся строкой [2..128] символов.

**Значение по умолчанию**

По умолчанию пароль равен логину.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-SUBSCRIBER-CONTROL

**Пример**

```
esr(config-subscriber-control)# session mac-authentication password userpassword
```



## session unknown-mac-address

Данной командой можно запретить прохождение трафика в аутентифицированной сессии для пакетов, у которых изменился MAC-адрес источника с момента аутентификации пользователя. Также при получении пакета с отличающимся MAC-адресом источника будет выведено сообщение в SYSLOG.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
[no] session unknown-mac-address filtering
```

### Параметры

Команда не содержит параметров.

### Значение по умолчанию

Прохождение трафика с отличающимся MAC-адресом источника разрешено.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG-SUBSCRIBER-CONTROL

### Пример

```
esr(config-subscriber-control)# session unknown-mac-address
```

## session-timeout

Данной командой задаётся интервал, по истечении которого, если не было пакетов от пользователя, сессия считается устаревшей и удаляется с устройства.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
session-timeout <SEC>
```

```
no session-timeout
```

### Параметры

<SEC> – период времени в секундах, принимает значения [120..3600].

### Значение по умолчанию

120 секунд.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-SUBSCRIBER-DEFAULT-SERVICE

**Пример**

```
esr(config-subscriber-default-service)# session-timeout 155
```

**show subscriber-control configuration**

Командой выполняется просмотр параметров конфигурации контроля пользователей.

**Синтаксис**

```
show subscriber-control configuration [ vrf <VRF> ]
```

**Параметры**

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример**

```
esr# show subscriber-control configuration
State:                Enabled
Description:          --
NAS IP address:       192.168.107.201
Sessions radius profile: RADIUS
Services radius profile: --
DAS profile:          bras2
Quota expired reauth: Disabled
Default service:
  Class map:          list1
  Filter name:        defaultserv
  Filter type:        local
  Filter action:      permit
  Default action:     redirect
  Default redirect URL: http://192.168.107.213:8080/eltex_portal/
```

**show subscriber-control radius-servers**

Данная команда используется для просмотра информации об используемых RADIUS-серверах.

## Синтаксис

```
show subscriber-control radius-servers [ vrf <VRF> ]
```

## Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут отображены используемые RADIUS-сервера в указанном VRF.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
esr# show subscriber-control radius-servers
IP address      Port      VRF         Usage         Connections count  Dead interval
Dead time
-----
-----
172.16.0.134    31813    --          services acct  0                  10
--
172.16.0.134    31812    --          services auth  0                  10
--
172.16.0.135    31813    --          sessions acct  0                  10
--
172.16.0.135    31812    --          sessions auth  0                  10
--
```

## show subscriber-control services count

Данные команды используются для просмотра информации и статистики по сервисам сессий контроля пользователей.

## Синтаксис

```
show subscriber-control services count [ { vrf <VRF> | service <SERVICE-NAME> } ]
```

## Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут отображены активные сервисы в указанном VRF.

<SERVICE-NAME> – имя сервиса, задаётся строкой до 220 символов.

## Необходимый уровень привилегий

1

**Командный режим**

ROOT

**Пример**

```

esr# show subscriber-control services count
Service           Active           Not active
-----
INTERNET          17              0
WELCOME           0               2

```

**show subscriber-control services counters**

Данные команды используются для просмотра счетчиков по сервисам сессий контроля пользователей.

**Синтаксис**

```

show subscriber-control services counters session-id <SESSION-ID> [ vrf <VRF> ]
[ service-id <SERVICE-ID> ] [ service <SERVICE-NAME> ]

```

**Параметры**

counters – просмотр статистики по сервису пользователя;

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут отображены активные сервисы в указанном VRF.

<SESSION-ID> – идентификатор сессии, принимает значения [1.. 18446744073709551615].

<SERVICE-ID> – идентификатор сессии, принимает значения [1.. 18446744073709551615].

<SERVICE-NAME> – имя сервиса, задаётся строкой до 220 символов.

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример**

```

esr# show subscriber-control services status counters session-id 5116089176692896081
Service id           Service name       Recv packets       Recv bytes          Send packets
Send bytes
-----
5152117973711859021 A INTERNET         2018               1615309             1761
281017

```

## show subscriber-control services status

Данные команды используются для просмотра оперативной информации по сервисам сессий контроля пользователей.

### Синтаксис

```
show subscriber-control services status [ session-id <SESSION-ID> ] [ { service-id <SERVICE-ID> | service <SERVICE-NAME> } ] [ vrf <VRF> ]
```

### Параметры

status – просмотр оперативной информации по сервису пользователя;

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут отображены активные сервисы в указанном VRF;

<SESSION-ID> – идентификатор сессии, принимает значения [1.. 18446744073709551615];

<SERVICE-ID> – идентификатор сессии, принимает значения [1.. 18446744073709551615];

<SERVICE-NAME> – имя сервиса, задаётся строкой до 220 символов.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
esr# show subscriber-control services status
Service id      Session id      Service name      User name      Quota volume      Quota time
-----
2522015791     2161727821     INTERNET5         79001110011   --                --
```

## show subscriber-control sessions count

Данные команды используются для просмотра информации и статистики по сессиям контроля пользователей.

### Синтаксис

```
show subscriber-control sessions count [ vrf <VRF> ]
```

### Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут отображены активные сессии в указанном VRF.

### Необходимый уровень привилегий

1

## Командный режим

ROOT

### Пример

```
esr# show subscriber-control sessions count
All:                18
Authenticated:     17
Not authenticated:  1
```

## show subscriber-control sessions counters

Данные команды используются для просмотра счетчиков по сессии пользователей.

### Синтаксис

```
show subscriber-control sessions counters { session-id <SESSION-ID> | username <SERVICE-NAME> } [ vrf <VRF> ]
```

### Параметры

status – просмотр оперативной информации по сессии пользователя;

counters – просмотр статистики по сессии пользователя;

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут отображены активные сессии в указанном VRF;

<SESSION-ID> – идентификатор сессии, принимает значения [1.. 18446744073709551615];

<USER-NAME> – имя пользователя, задаётся строкой до 230 символов.

### Необходимый уровень привилегий

1

## Командный режим

ROOT

### Пример

```
esr# show subscriber-control sessions counters session-id 5116089176692896081
User name          Recv packets      Recv bytes        Send packets      Send bytes
-----
79231552525       2198              1697762           1984              313452
```

## show subscriber-control sessions status

Данные команды используются для просмотра оперативной информации по сессии пользователей.

## Синтаксис

```
show subscriber-control sessions status [ { session-id <SESSION-ID> | username <SERVICE-NAME> } ] [ vrf <VRF> ]
```

## Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут отображены активные сессии в указанном VRF;

<SESSION-ID> – идентификатор сессии, принимает значения [1.. 18446744073709551615];

<USER-NAME> – имя пользователя, задаётся строкой до 230 символов.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
esr# show subscriber-control sessions status
Session id      User name      IP address      MAC address      Interface      Domain
-----
2161727821     79001110011   192.168.244.12  c4:12:f5:d4:af:70  bridge 13     root
esr# show subscriber-control sessions counters session-id 2161727821
User name      Recv packets   Recv bytes      Send packets     Send bytes
-----
79001110011   243            87056           294              35961
```

## subscriber-control

Команда используется для создания профиля контроля пользователей и перехода в режим его конфигурирования.

Использование отрицательной формы команды (no) удаляет указанный профиль контроля пользователей.

## Синтаксис

```
[no] subscriber-control [ vrf <VRF> ]
```

## Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать контроль пользователей.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

### Пример

```
esr(config)# subscriber-control
```

## subscriber-control application-filter

Данная команда используется для включения контроля приложений на интерфейсе.

Использование отрицательной формы команды (no) выключает контроль приложений на интерфейсе.

### Синтаксис

```
subscriber-control application-filter <NAME>
```

```
no subscriber-control application-filter
```

### Параметры

<NAME> – имя профиля приложений, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

15

## Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-L2TPV3

### Пример

```
esr(config-if-gi)# subscriber-control application-filter LIST
```



## subscriber-control apps-server-url

Данная команда используется для задания URL-адреса сервера, предоставляющего списки приложений для фильтрации трафика. Списки запрашиваются у сервера в момент аутентификации пользователя.

Использование отрицательной формы команды (no) удаляет заданный URL-сервера.

### Синтаксис

```
subscriber-control apps-server-url <URL>  
no subscriber-control apps-server-url
```

### Параметры

<URL> – адрес ссылки, задаётся строкой от 8 до 255 символов.

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

15

### Командный режим

CONFIG

### Пример

```
esr(config)# subscriber-control apps-server-url "http://192.168.1.1/files/"
```

## subscriber-control filters-server-url

Данная команда используется для задания адреса сервера, предоставляющего списки URL для фильтрации HTTP/HTTPS-трафика. Списки запрашиваются у сервера в момент аутентификации пользователя.

Использование отрицательной формы команды (no) удаляет заданный URL-сервера.

### Синтаксис

```
subscriber-control filters-server-url <URL>  
no subscriber-control filters-server-url
```

### Параметры

<URL> – адрес ссылки, задаётся строкой до 255 символов.

### Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# subscriber-control filters-server-url "http://192.168.1.1/files/"
```

## subscriber-control peer-address

Данной командой определяется IP-адрес соседа, с которым будет осуществляться синхронизация таблицы с option 82.

Использование отрицательной формы команды (no) удаляет IP-адрес соседнего маршрутизатора из конфигурации.

## Синтаксис

```
subscriber-control peer-address <ADDR>  
no subscriber-control peer-address
```

## Параметры

<ADDR> – IP-адрес соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG

## Пример

```
esr(config)# subscriber-control peer-address 10.24.80.100
```

## subscriber-control thresholds sessions-number

Данная команда используется для задания порога количества сессий BRAS всех профилей контроля пользователей для отправки snmp-trap eltexBrasSessionsNumberHigh и eltexBrasSessionsNumberHighOk.

Использование отрицательной формы команды (no) устанавливает значения по умолчанию.

**Синтаксис**

```
subscriber-control thresholds sessions-number { high <TH-HIGH> | low <TH-LOW> }
no subscriber-control thresholds sessions-number { high | low }
```

**Параметры**

<TH-HIGH> – порог количества сессий BRAS для отправки snmp-trap eltexBrasSessionsNumberHigh;

<TH-LOW> – порог количества сессий BRAS для отправки snmp-trap eltexBrasSessionsNumberHighOk.

**Значение по умолчанию**

На ESR-1700 <TH-HIGH> – 47000, <TH-LOW> – 46000;

На ESR-1000/1200/1500/1511/3100/3200 <TH-HIGH> – 9000, <TH-LOW> – 8500;

На ESR-100/200 <TH-HIGH> – 900, <TH-LOW> – 850.

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG

**Пример**

```
esr(config)# subscriber-control thresholds sessions-number high 8000
```

**subscriber-control unused-filters-remove-delay**

Данная команда используется для задания интервала, по истечении которого с устройства будут удалены неиспользуемые в текущий момент списки URL.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

```
subscriber-control unused-filters-remove-delay <DELAY>
no subscriber-control unused-filters-remove-delay
```

**Параметры**

<DELAY> – временной интервал в секундах, принимает значения [10800..86400].

**Значение по умолчанию**

10800

**Необходимый уровень привилегий**

15

## Командный режим

CONFIG

### Пример

```
esr(config)# subscriber-control unused-filters-remove-delay 40000
```

## thresholds sessions-number

Данная команда используется для задания порога количества сессий BRAS для отправки snmp-trapeltexBrasSessionsNumberHigh и eltexBrasSessionsNumberHighOk.

Использование отрицательной формы команды (no) устанавливает значения по умолчанию.

### Синтаксис

```
thresholds sessions-number { high <TH-HIGH> | low <TH-LOW> }
no thresholds sessions-number { high | low }
```

### Параметры

<TH-HIGH> – порог количества сессий BRAS для отправки snmp-trap eltexBrasSessionsNumberHigh;

<TH-LOW> – порог количества сессий BRAS для отправки snmp-trapeltexBrasSessionsNumberHighOk.

### Значение по умолчанию

На ESR-1700 <TH-HIGH> – 47000, <TH-LOW> – 46000;

На ESR-1000/1200/1500/1511/3100 <TH-HIGH> – 9000, <TH-LOW> – 8500;

На ESR-20/21/30/100/200 <TH-HIGH> – 900, <TH-LOW> – 850.

### Необходимый уровень привилегий

15

## Командный режим

CONFIG-SUBSCRIBER-CONTROL

### Пример

```
esr(config-subscriber-control)# thresholds sessions-number high 8000
```

## vrrp-group

Данной командой определяется VRRP-группа, на основе которой определяется состояние (основной/резервный) сервиса контроля абонентов. При переключении VRRP в состояние BACKUP происходит сброс всех сессий контроля пользователей.

Использование отрицательной формы команды (no) удаляет идентификатор VRRP.

**Синтаксис**

```
vrrp-group <GRID>  
no vrrp-group
```

**Параметры**

<GRID> – идентификатор группы VRRP-маршрутизатора, принимает значения [1..32].

**Необходимый уровень привилегий**

15

**Командный режим**

CONFIG-SUBSCRIBER-CONTROL

**Пример**

```
esr(config-subscriber-control)# vrrp-group 10
```

## 32 Настройка SoftGRE контроллера туннелей

**!** В текущей версии ПО данный функционал поддерживается только на маршрутизаторах ESR-15/15R/30/100/200/1000/1200/1500/1511/1700/3200 по лицензии. Для контроллера WLC лицензия на сервис встроена.

- [aaa das-profile](#)
- [aaa radius-profile](#)
- [data-tunnel configuration](#)
- [enable](#)
- [failover](#)
- [failure-count](#)
- [internal-usage-vlan](#)
- [keepalive mode](#)
- [keepalive-disable](#)
- [nas-ip-address](#)
- [resp-time](#)
- [retry-time](#)
- [service-vlan](#)
- [show interfaces bridge switch-communities](#)
- [show interfaces bridge switch-communities community](#)
- [show interfaces bridge switch-communities community include](#)
- [show interfaces bridge switch-communities summary](#)
- [show softgre-controller access-points](#)
- [thresholds tunnels-number](#)
- [thresholds sub-tunnels-number](#)
- [tunnel-isolation](#)
- [softgre-controller](#)

### aaa das-profile

Данная команда используется для выбора профиля серверов динамической авторизации (DAS), на которые будут приходиться CoA-запросы от PCRF об изменении политики обслуживания.

Использование отрицательной формы команды (no) удаляет заданный профиль серверов динамической авторизации (DAS).

#### Синтаксис

```
[no] aaa das-profile <NAME>
```

#### Параметры

<NAME> – имя профиля серверов динамической авторизации (DAS), задается строкой до 31 символа.

#### Необходимый уровень привилегий

15

#### Командный режим

CONFIG-SOFTGRE-CONTROLLER

## Пример

```
esr(config-softgre-controller)# aaa das-profile profile1
```

## aaa radius-profile

Данная команда используется для выбора профиля RADIUS-серверов, на которые будут отправляться запросы для получения конфигурации SoftGRE DATA-туннелей (обслуживаемых SSID и параметров шейпинга).

Использование отрицательной формы команды (no) удаляет заданный профиль RADIUS-серверов.

## Синтаксис

```
[no] aaa radius-profile <NAME>
```

## Параметры

<NAME> – имя профиля RADIUS-серверов, задается строкой до 31 символа.

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-SOFTGRE-CONTROLLER

## Пример

```
esr(config-softgre-controller)# aaa radius-profile profile1
```

## data-tunnel configuration

Данной командой устанавливается режим конфигурации SoftGRE DATA туннелей.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
data-tunnel configuration { local | radius | wlc }
no data-tunnel configuration
```

## Параметры

local – режим конфигурации, при котором параметры SoftGRE DATA туннелей получают из локальной конфигурации маршрутизатора;

radius – режим, при котором параметры SoftGRE DATA туннелей запрашиваются у RADIUS-сервера;

wlc – режим, при котором туннелями SoftGRE DATA управляет WLC.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-SOFTGRE-CONTROLLER

## Пример

```
esr(config-softgre-controller)# data-tunnel configuration wlc
```

## enable

Данной командой активируется контроллер Wi-Fi.

Использование отрицательной формы команды (no) деактивирует контроллер Wi-Fi.

## Синтаксис

[no] enable

## Параметры

Команда не содержит параметров.

## Значение по умолчанию

Процесс выключен.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-SOFTGRE-CONTROLLER

## Пример

```
esr(config-softgre-controller)# enable
```

## failover

Данная команда активирует работу резервирования туннелей. Указание адресации и группы описано в разделе [Резервирование](#).

Использование отрицательной формы команды (no) выключает работу резервирования туннелей.

Синтаксис

[no] failover



## Параметры

Отключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WIRELESS

## Пример

```
wlc(config-wireless)# failover
```

## failure-count

Данной командой определяется количество последовательных неудачных ICMP-запросов, после которых, при отсутствии ответа от встречной стороны, туннель считается нерабочим и удаляется из системы.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
[no] failure-count <VALUE>
```

## Параметры

<VALUE> – количество неудачных ping-запросов, принимает значения [1..100].

## Значение по умолчанию

5

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-SOFTGRE-CONTROLLER

## Пример

```
esr(config-softgre-controller)# failure-count 8
```

## internal-usage-vlan

Данной командой определяется список VLAN, которые будут использоваться для формирования switch-community.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
[no] internal-usage-vlan <ACT> <VID>
```

## Параметры

<ACT> – назначаемое действие:

- add – включение интерфейса во VLAN;
- remove – исключение интерфейса из VLAN.

<VID> – идентификационный номер VLAN, задаётся в диапазоне [2...4094]. Можно задать диапазоном через «-» или перечислением через «,».

## Значение по умолчанию

Отсутствует.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-SOFTGRE-CONTROLLER

## Пример

```
esr(config-softgre-controller)# internal-usage-vlan add 1001-2037,2500-3000,3200-3700
```

## keepalive mode

Данной командой устанавливается режим работы механизма keepalive для SoftGRE-туннелей.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

```
keepalive mode { proactive | reactive }
no keepalive mode
```

## Параметры

- proactive – режим проверки SoftGRE-туннелей через icmp-проверку удаленного адреса;
- reactive – режим проверки SoftGRE-туннелей через прием icmp-пакетов с payload: "ELTEX\_GRE IPv4 < туннельный ip address устройства>" от удаленных устройств внутри туннеля.

## Значение по умолчанию

proactive

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-SOFTGRE-CONTROLLER

## Пример

```
esr(config-softgre-controller)# keepalive mode reactive
```

## keepalive-disable

Данная команда отключает обмен ICMP-сообщениями, которые используются для проверки доступности удаленного шлюза туннелей Wi-Fi контроллера.

Использование отрицательной формы команды (no) включает обмен ICMP-сообщениями.

## Синтаксис

[no] keepalive-disable

## Параметры

Команда не содержит параметров.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-SOFTGRE-CONTROLLER

## Пример

```
esr(config-softgre-controller)# keepalive-disable
```

## nas-ip-address

Данной командой определяется IP-адрес маршрутизатора, который будет использоваться в качестве IP-адреса источника в отправляемых RADIUS-пакетах.

Использование отрицательной формы команды (no) удаляет указанный IP-адрес источника.

## Синтаксис

nas-ip-address <ADDR>

no nas-ip-address

## Параметры

<ADDR> – IP-адрес источника, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

## Необходимый уровень привилегий

15

## Командный режим

CONFIG-SOFTGRE-CONTROLLER

## Пример

```
esr(config-softgre-controller)# nas-ip-address 10.100.100.2
```

## resp-time

Данной командой определяется время ожидания ответа, после истечения которого ICMP-запрос считается отклоненным.

Использование отрицательной формы устанавливает значение resp-time по умолчанию.

## Синтаксис

[no] resp-time <TIME >

## Параметры

<TIME > – количество секунд, принимает значения [1..30].

## Значение по умолчанию

5

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-SOFTGRE-CONTROLLER

## Пример

```
esr(config-softgre-controller)# resp-time 30
```

## retry-time

Данной командой устанавливается время между ICMP-запросами.

Использование отрицательной формы команды (no) устанавливает значение retry-time по умолчанию.

### Синтаксис

```
[no] retry time < TIME >
```

### Параметры

< TIME > – количество секунд, принимает значения [60..3600].

### Необходимый уровень привилегий

10

### Значение по умолчанию

60

### Командный режим

CONFIG-SOFTGRE-CONTROLLER

### Пример

```
esr(config-softgre-controller)# retry-time 12
```

### service-vlan

Данной командой определяется список VLAN, которые будут включаться в SoftGRE-туннели в режиме конфигурирования data-tunnel configuration wlc.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
[no] service-vlan <ACT> <VID>
```

### Параметры

<ACT> – назначаемое действие:

- add – включение интерфейса во VLAN;
- remove – исключение интерфейса из VLAN.

<VID> – идентификационный номер VLAN, задаётся в диапазоне [2...4094]. Можно задать диапазоном через «-» или перечислением через «,».

### Значение по умолчанию

Отсутствует.

### Необходимый уровень привилегий

10

**Командный режим**

CONFIG-SOFTGRE-CONTROLLER

**Пример**

```
esr(config-softgre-controller)# service-vlan add 1001-2037,2500-3000,3200-3700
```

**show interfaces bridge switch-communities**

Командой выполняется просмотр списка сформированных switch-community.

**Синтаксис**

```
show interfaces bridge switch-communities [<BRIDGE-ID>]
```

**Параметры**

<BRIDGE-ID> – идентификационный номер моста, задается в виде, описанном в разделе [Типы и порядок именования интерфейсов маршрутизатора](#).

**Необходимый уровень привилегий**

1

**Командный режим**

ROOT

**Пример**

```
esr# show interfaces bridge switch-communities
bridge 10
Location                               Community                               Interfaces
-----                               -
data10                                  Eltex.Novosibirsk.Novosibirskaya_oblast  softgre 2.10,
softgre 4.10, softgre 6.10,
.MRF_Sibir.Local.AllClients.root         softgre 8.10
data10                                  TEST.Novosibirsk.Novosibirskaya_oblast  softgre 2.310,
softgre 4.310, softgre
.MRF_Sibir.Local.AllClients.root         6.310, softgre
8.310
bridge 11
Location                               Community                               Interfaces
-----                               -
data11                                  Eltex.Novosibirsk.Novosibirskaya_oblast  softgre 2.311,
softgre 4.311, softgre
.MRF_Sibir.Local.AllClients.root         6.311, softgre
8.311
```

## show interfaces bridge switch-communities community

Командой выполняется просмотр списка сформированных switch-community, соответствующих определенному названию домена СУ EMS.

### Синтаксис

```
show interfaces bridge switch-communities community <EMS-DOMAIN> [<BRIDGE-ID>]
```

### Параметры

<EMS-DOMAIN> – имя домена СУ EMS, WORD(1-246).

<BRIDGE-ID> – идентификационный номер моста, задается в виде, описанном в разделе [Типы и порядок именованя интерфейсов маршрутизатора](#).

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```

esr# show interfaces bridge switch-communities community
Eltex.Novosibirsk.Novosibirskaya_oblast.MRF_Sibir.Local.AllClients.root
bridge 10
Location                               Community                               Interfaces
-----                               -
data10                                  Eltex.Novosibirsk.Novosibirskaya_oblast softgre 2.10,
softgre 4.10, softgre 6.10,
.MRF_Sibir.Local.AllClients.root       softgre 8.10

bridge 11
Location                               Community                               Interfaces
-----                               -
data11                                  Eltex.Novosibirsk.Novosibirskaya_oblast softgre 2.311,
softgre 4.311, softgre
.MRF_Sibir.Local.AllClients.root       6.311, softgre
8.311

```

## show interfaces bridge switch-communities community include

Командой выполняется просмотр списка сформированных switch-community, соответствующих определенному шаблону в названии домена СУ EMS.

### Синтаксис

```
show interfaces bridge switch-communities community include <WORD> [<BRIDGE-ID>]
```

## Параметры

<WORD> – шаблон, по которому выполняется поиск в имени домена СУ EMS, WORD(1-246).

<BRIDGE-ID> – идентификационный номер моста, задается в виде, описанном в разделе [Типы и порядок именованя интерфейсов маршрутизатора](#).

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```

esr# show interfaces bridge switch-communities community include eltex
bridge 10
Location                Community                Interfaces
-----
data10                  Eltex.Novosibirsk.Novosibirskaya_oblast softgre 2.10,
softgre 4.10, softgre 6.10,
                        .MRF_Sibir.Local.AllClients.root      softgre 8.10

bridge 11
Location                Community                Interfaces
-----
data11                  Eltex.Novosibirsk.Novosibirskaya_oblast softgre 2.311,
softgre 4.311, softgre
                        .MRF_Sibir.Local.AllClients.root      6.311, softgre
8.311

```

## show interfaces bridge switch-communities summary

Командой выполняется просмотр количества сформированных switch-community.

## Синтаксис

```
show interfaces bridge switch-communities summary [<BRIDGE-ID>]
```

## Параметры

<BRIDGE-ID> – идентификационный номер моста, задается в виде, описанном в разделе [Типы и порядок именованя интерфейсов маршрутизатора](#).

## Необходимый уровень привилегий

1



## Командный режим

ROOT

### Пример

```

esr# show interfaces bridge switch-communities summary
Bridges      Communities
-----
bridge 10    2
bridge 11    1

```

## show softgre-controller access-points

Команда вводит суммирующую информацию о подключенных ТД:

- MAC-адрес;
- туннельный адрес;
- адрес управления;
- сабтуннели.

### Синтаксис

```

show softgre-controller access-points [ mac <ADDR> | management-ip <ADDR> | remote-ip <ADDR> ]

```

### Параметры

mac <ADDR> – MAC-адрес ТД, по которому ведется поиск, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF];

management-ip <ADDR> – IP-адрес управления, по которому ведется поиск, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

remote-ip <ADDR> – IP-адрес, поднимающий туннель, по которому ведется поиск, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

### Необходимый уровень привилегий

1

## Командный режим

ROOT

### Пример

```

esr# show softgre-controller access-points
MAC address      Remote IP      Management IP      Sub tunnels
-----
e0:d9:e3:73:07:61  100.124.0.36  10.102.14.216  softgre 1.1, softgre 2.12, softgre
2.14
a8:f9:4b:b0:2c:c1  100.124.0.34  10.102.29.229  softgre 3.1, softgre 4.12, softgre
4.122, softgre 4.321, softgre 4.322,

```

```

softgre
a8:f9:4b:ac:ae:aa 100.124.0.231 10.102.16.73 softgre 4.323, softgre 4.489,
4.2408
softgre
00:00:2a:d9:d8:a7 10.10.45.2 10.102.17.248 softgre 5.1, softgre 6.12, softgre
6.122, softgre 6.321, softgre 6.322,
softgre 6.323, softgre 6.489,
6.2408
00:00:2a:d9:d8:a5 10.10.45.1 10.102.18.115 softgre 7.1, softgre 8.12, softgre
8.2011, softgre 8.2013
00:00:2a:d9:d8:a9 10.10.45.3 10.102.18.196 softgre 9.1, softgre 10.12, softgre
10.2011, softgre 10.2013
softgre 11.1, softgre 12.12, softgre
12.2011, softgre 12.2013

```

```

esr# show softgre-controller access-points remote-ip 100.124.0.231
MAC adress          Remote IP           Management IP       Sub tunnels
-----
a8:f9:4b:ac:ae:aa  100.124.0.231     10.102.16.73      softgre 5.1, softgre 6.12, softgre
6.122, softgre 6.321, softgre 6.322,
softgre 6.323, softgre 6.489,
softgre
6.2408

```

## thresholds tunnels-number

Данная команда используется для задания порога количества туннелей SoftGRE для отправки snmp-trap eltexInterfacesNumberHigh и eltexInterfacesNumberHighOk.

Использование отрицательной формы команды (no) устанавливает значения по умолчанию.

### Синтаксис

```

thresholds tunnels-number { high <TH-HIGH> | low <TH-LOW> }
no thresholds tunnels-number { high | low }

```

### Параметры

<TH-HIGH> – порог количества туннелей SoftGRE для отправки snmp-trap eltexInterfacesNumberHigh;

<TH-LOW> – порог количества туннелей SoftGRE для отправки snmp-trap eltexInterfacesNumberHighOk.

### Значение по умолчанию

На ESR-1000 <TH-HIGH> – 900, <TH-LOW> – 850

На ESR-100 и ESR-200 <TH-HIGH> – 400, <TH-LOW> – 350

### Необходимый уровень привилегий

10

**Командный режим**

CONFIG-SOFTGRE-CONTROLLER

**Пример**

```
esr(config-softgre-controller)# thresholds tunnels-number high 1000
```

**thresholds sub-tunnels-number**

Данная команда используется для задания порога количества сабтуннелей SoftGRE для отправки snmp-trap eltexInterfacesNumberHigh и eltexInterfacesNumberHighOk.

Использование отрицательной формы команды (no) устанавливает значения по умолчанию.

**Синтаксис**

```
thresholds sub-tunnels-number { high <TH-HIGH> | low <TH-LOW> }
```

```
no thresholds sub-tunnels-number { high | low }
```

**Параметры**

<TH-HIGH> – порог количества сабтуннелей SoftGRE для отправки snmp-trap eltexInterfacesNumberHigh;

<TH-LOW> – порог количества сабтуннелей SoftGRE для отправки snmp-trap eltexInterfacesNumberHighOk.

**Значение по умолчанию**

На ESR-1700, ESR-1200 <TH-HIGH> – 7000, <TH-LOW> – 7500;

На ESR-3100 <TH-HIGH> – 7800, <TH-LOW> – 7600;

На ESR-1000 <TH-HIGH> – 2500, <TH-LOW> – 2000;

На ESR-100 и ESR-200 <TH-HIGH> – 1000, <TH-LOW> – 900.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-SOFTGRE-CONTROLLER

**Пример**

```
esr(config-softgre-controller)# thresholds sub-tunnels-number high 2000
```

**tunnel-isolation**

Данная команда включает изоляцию между клиентами в разных SoftGRE-туннелях.

Использование отрицательной формы команды (no) отключает изоляцию между клиентами в разных SoftGRE-туннелях.

### Синтаксис

```
[no] tunnel-isolation
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-SOFTGRE-CONTROLLER

### Пример

```
esr(config-softgre-controller)# tunnel-isolation
```

## softgre-controller

Переход в режим конфигурирования контроллера SoftGRE.

Использование отрицательной формы команды (no) очищает конфигурацию и выключает контроллер SoftGRE туннелей.

### Синтаксис

```
[no] softgre-controller
```

### Параметры

Команда не содержит параметров.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# softgre-controller
```

## ТЕХНИЧЕСКАЯ ПОДДЕРЖКА

Для получения технической консультации по вопросам эксплуатации оборудования ООО «Предприятие «ЭЛТЕКС» Вы можете обратиться в Сервисный центр компании:

Форма обратной связи на сайте: <https://eltex-co.ru/support/>

Servicedesk: <https://servicedesk.eltex-co.ru>

На официальном сайте компании Вы можете найти техническую документацию и программное обеспечение для продукции ООО «Предприятие «ЭЛТЕКС», обратиться к базе знаний и оставить интерактивную заявку:

Официальный сайт компании: <https://eltex-co.ru>

База знаний: <https://docs.eltex-co.ru/display/EKB/Eltex+Knowledge+Base>

Центр загрузок: <https://eltex-co.ru/support/downloads>